An Approach to Traffic Management in a Navigational Channel

by

Roberto E. Soto

B.S., Civil Engineering, Universidad del Zulia (1989)

Submitted in Partial Fulfillment
of the Requirements for the Degree of

MASTER OF SCIENCE
in Ocean Engineering

at the

Massachusetts Institute of Technology

May 1995

Signature of Author...
Department of Ocean Engineering
May, 1995

Certified by..
Dr. Ernst G. Frankel
Professor of Marine Systems
Thesis Supervisor

Accepted by ....
Dr. A. Douglas Carmichael
Chairman, Departmental Graduate Committee
May, 1995

An Approach to Traffic Management

in a Navigational Channel

by

Roberto E. Soto

Submitted in partial fulfillment of the requirements for

the Degree of MASTER OF SCIENCE

May 1995

Abstract

A systemic approach to traffic control on a navigation channel is proposed. A general review of navigation channel geometry and practices is conducted. Factors that affect traffic are evaluated. A model for traffic control and management is proposed. This model allows for traffic simulation (with user provided data) and modeling (with data generated by the model). Parameters in the model can be changed to provide the user with an overall best strategy for traffic management. Results and applicability of the model are discussed. A system wide approach to traffic management is then defined using the proposed model and existing computing and communications technology.

Thesis Supervisor: Ernst G. Frankel, Ph.D.,D.B.A.
Title: Professor of Marine Systems

CONTENTS

List of Figures

List of Tables

CHAPTER No. 1

1. INTRODUCTION

Navigation channels provide the access to ports from the open sea. Their dimensions decide the size of ships that can approach a terminal and consequently affect the capacity of the terminal.

Most channels are not natural but dredged so they require periodic investment to maintain the water depth required for safe passage of ships. Periodic dredging usually is a large part of a port's operating budget, running sometimes into the millions of dollars annually. Therefore, it makes sense to minimize dredging costs by allowing the largest possible number of ships into the terminal with the smallest dredging section possible. Of course, safety considerations limit the smallest dredging section. This is usually considered by port managers and designers. They do not, however, weight in the fact that through effective traffic management, the capacity of a channel can be markedly improved.

In chapter No. 2, channel dimensions and their effect on navigation will be discussed. Changes in dimensions that could be carried out under a traffic control system are defined. The idea is that a traffic management system could, conceivably, not only increase the number of ships using a channel but also allow for greater safety. Therefore, a part of the chapter is dedicated to safety issues in navigation and the important stopping distance.

It has been proven that a channel's capacity is primarily influenced by the queuing model used in traffic management. In chapter No. 3, a model is proposed to simulate and model traffic control in a channel. The model should allow a user to decide the most efficient way under his guidelines. Three queuing modes are incorporated in the model. Among these, most ports use a first come, first based model (FCFS) in which whoever gets near the entrance first gets into the terminal first. Other strategies allow the costliest ship (COST) or the fastest ship (FAST) to go first. The merits of each of these strategies in traffic control are discussed. Other factors influencing capacity are also included to find their effect on traffic.

Chapter 4 is dedicated to economic impact of alternatives. The effect of a traffic control system with a queuing model is estimated. A best strategy under each different circumstance is proposed.

A technological feasible traffic management system is described in chapter No. 5. The system's requirements and operation are defined by the channel dimensions changes proposed in chapter 2, by the results obtained with traffic simulation in chapter 3 and the most cost-effective strategy of chapter 4. Other existing traffic systems are described, as is their influence on traffic improvements.

Finally chapter No. 6, includes conclusions and recommendations.

CHAPTER No. 2

2. THE NAVIGATION CHANNEL

Since the final objective is a traffic management system, channel design and dimensions will be considered under the light of their effect on capacity. Here, emphasis will be placed on those factors in channel dimensions that influence capacity.

When designing a channel, engineers consider factors such as design ship, safety, sedimentation rates, wind speed and direction, current velocity and direction, dredge availability and of course, required capacity. These factors can be roughly divided into to categories: one concerning the safe passage of the ship and the other related to economic considerations.

2.1 DESIGN

The size of the largest ship that will traverse a channel and the characteristics of the area in which the channel will be built dictates its dimensions. However, the multitude of factors affecting the engineering decision has made a clear definition of optimum channel dimensions elusive. So safety is the primary consideration in design. The problem is that, in navigation channels, large factors of security result in high construction and maintenance costs.

2.1.1 Channel width

The nominal width over which the nominal depth exists defines the channel width. ICORELS of PIANC (International Committee for the Reception of Large Vessels of the Permanent International Association of Navigation Congress, Ref. 17) recommends channel width to be a function of:

(a) Maximum beam of vessels to be received
(b) Difference between the vessel's true position and that estimated by the mariner
(c) Additional deviation that might occur from the moment when de deviation is first noted to the moment when the correction becomes effective.
(d) Additional width needed to account for drifting due to cross currents and cross winds.
(e) A safety margin not less than half the beam of the vessel on each side.

ICORELS says that "the present state of knowledge of the influence of the various parameters mentioned above on the necessary nominal width of access channels can only be partly assessed with de desired accuracy...Standard widths of channels in sea straits cannot be established. They are a compromise between several factors, depending mainly in the intensity of traffic, the possibility of a sound layout of the channel in the sea strait, on the conditions of tide and weather, and on the dimensions of ships that can traverse the least depth of the strait."

However, the Commission considered that the nominal width should be no less than five times the beam of the largest vessel for a one way straight channel. Curves have to be considered separately. In two-way channels, passing of ships in not dangerous if the distance apart is at least twice the beam of the larger vessel accounting for limited accuracy of passing maneuvers. These suggestions by PIANC are based on observations made in various ports. The results were so varied that the commission could not conclude on a definitive recommendation. This is probably due to the various factors influencing the behavior of a ship on a channel. ICORELS mentions drifting due to cross winds and currents as affecting the width selection. Also, channel geometry itself plays a role in ship control. Relative depth, lateral slopes and width all affect the way a ship behaves on a navigation channel.

Models of ships behavior in canals (Delft Hydraulics, Ref. 6) suggest that channel width greatly affects stopping distance, an important factor to be considered later. The report states the stopping distance "has to be defined as a function of the maneuvering space available to the ship on either side... the width factor." It recommends a width of 4.5 to 5 times the beam of the ship for easy maneuvering.

Kray (Ref. 14) has several studies in which he assesses the effect of channel slope in ship maneuverability. His findings suggest that gentler channel slopes result in increased ship maneuverability. Other publications (Ref. 2, 8 and 17) show that the ratio of a ship's draft to water depth also influences the ship's maneuverability and therefore channel width.

It should be noted that experiences in Germany (Ref. 11), in which deviation from a center line was measured, show that the lane occupied by one-way traffic on long straight lines has an average width between 3.6 and 4.5 times the ship's beam. The bigger coefficient refers to outbound traffic with less draft. These experiences correspond to a channel with a nominal width of 3.5 the ship's beam restricted by narrow banks on both sides. Other experiences suggest controlled traffic deviates less from a center line than non controlled traffic (Ref. 10). Unfortunately, these experiences do not result in a recommendation for channel width under controlled traffic conditions.

Thorensen (Ref. 21) proposes a total channel bottom for single lane channels of 3.6 to six times the beam of the design ship, the selection dependent on sea and wind conditions and soil type. Sea and wind conditions define the width of the maneuvering lane. This author sets a maneuvering factor between 1.6 and two times the beam of the ship. Bank clearance counteracts the effect of bank suction if it is set at one to two times the beam of the largest ship, one corresponding to a gentler side slope.

For two way channels, Thorensen suggests a total width between 6.2 and nine times the beam of the design ship. He bases his advice on the same factors outlined before and includes a ship clearance lane of 30 m. or the beam of the largest ship. Note that Thorensen's recommendations follow the field data obtained by ICOREL.

Since the introduction of traffic management is likely to reduce the necessary navigation portion of channel width, three different scenarios representing different channel widths will be considered. Details about the navigation system that should support each scenario will be discussed in chapter No. 5.

The first scenario will assume that the maneuvering factor is the largest one, two (fig. 2.1-a). This scenario will serve as a base for comparison. This corresponds to the actual design condition of a navigation channel. The resulting width is five times the beam for a one-way channel and eight times the beam for a two-way channel.

The second scenario will presume the existence of improved navigation control. Therefore, the maneuvering lane reduces to 1.6 times the beam of the design ship without loss in safety (Fig 2.1-b). This is justified if a navigation system provides accurate and continuous information on position to the navigator. Now, the width will be 4.6 times the beam of the design ship for a single lane channel and 7.2 times the beam for a double lane channel.

TABLE No. 2.1

CHANNEL WIDTH

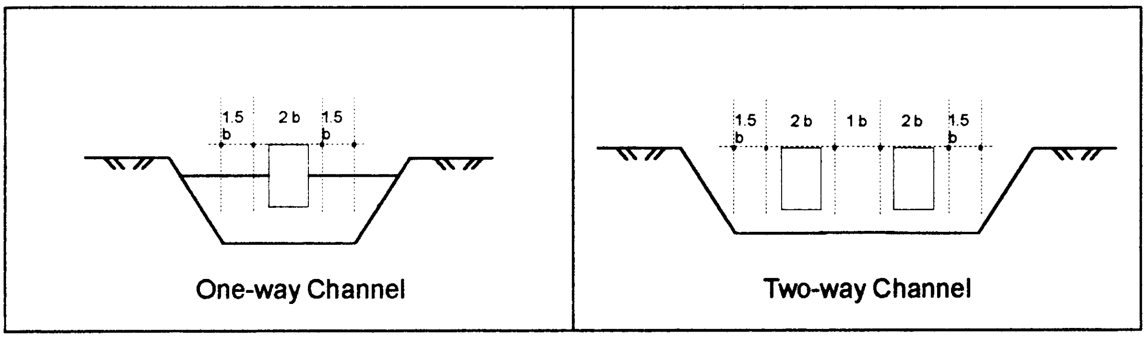
Scenario	One - way Channel	Two - way Channel
1	5.0 b	8.0 b
2	4.6 b	7.2 b
3	4.0 b	6.0 b

Note: b stands for vessel beam.

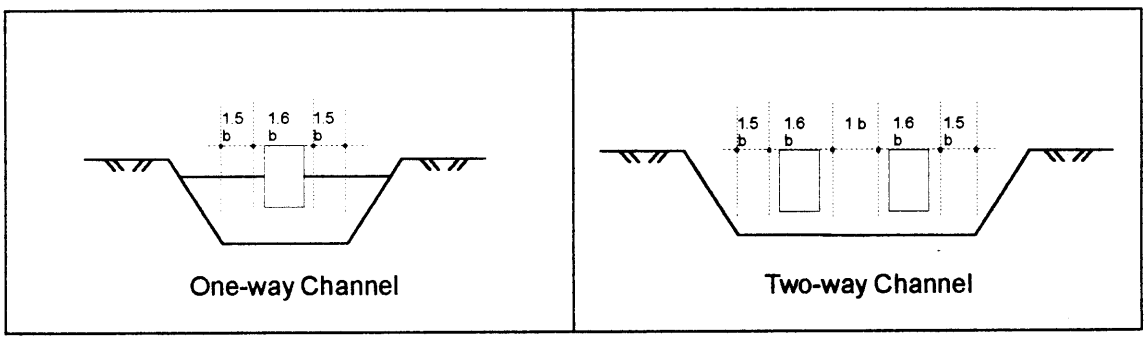
A third scenario is a "what if" scenario. Assume for a moment a system is set up so the ship relinquishes control to an outside mechanism that can guide the ship toward the terminal. Such system would eliminate the need for a maneuvering lane and therefore, the maneuvering lane factor would

Figure No. 2.1
CHANNEL WIDTH

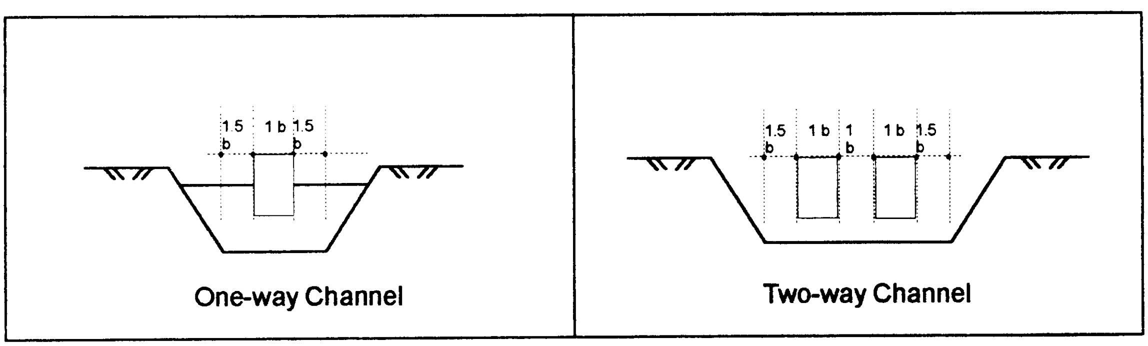
Scenario A



Scenario B



Scenario C



b: design ship beam

be one times the beam of the design ship (Fig. 2.1-c). In this scenario, two-way channels will require six beams as width while one-way channel will require four beams. Channel widths are tabulated in table No. 2.1 for the outlined scenarios.

2.1.2 Channel depth

The depth of a navigation channel affects the safety of the operation, the capacity of the channel and its maintenance requirements. A shallow channel "causes an increase in the ship's required driving power and impairs controllability in maneuvering the ship.. because of slow rudder response" (Kray, Ref. 14). Shallow waters are defined as those in which the depth to draft ratio is less than two.

Kray also observes that ship traveling through a shallow channel may cause movement of soft bottom material. This might result on an undesirable redistribution from a maintenance point of view (The effects of depth in channel capacity and maintenance will be further discussed in point 2.2.). Thorensen (Ref. 21) points out that bank suction increases when underkeal clearance decreases.

Different authors advocate different approaches to depth determination. The most general recommendation (Ref. 5) is to have a ratio of channel depth to mean draft of the largest vessel between 1.3 and 1.5. For ships moving at speeds below six knots, a lower figure can be used (Kray, Ref. 14). The selection between 1.3 and 1.5 will depend on factors such as:

- Draft of design vessel
- Tidal variations
- Wave-ship motions
- Load ship motions
- Wind effect on water level and on tides
- Bottom suction
- Water density
- Increased sinkage because of passing vessels
- Increased sinkage away from channel centerline

PIANC (Ref. 18) recommends a depth determined from the following formula:

$$h = D+L+k+Z+I+R+C+IWL \qquad \text{Eq. 2.1}$$

where,

h: channel depth in relation to the level of the marine chart,

D: draft of a stationary vessel in slack/still water,

L: deviation from above chart datum of the hydrographic chart,

Z: squat and trim,

I: amplitude of vertical ship movement,

R: bottom roughness,

C: keel clearance and

IWL: inaccuracy of water-level measurement and sounding.

The previous formula does not consider sedimentary processes, assuming no siltation after initial construction. However, sedimentary process are important in most channels and should be considered when defining navigation depth.

Note that squat becomes important when large vessels are involved. Figure 2.2 shows data obtained by the National Hydraulic Lab, France for a 250.000 dwt., 20 m. draft vessel with a bulbous bow model for different under keel clearances (Ref. 8). This and other laboratory experiments resulted in the following formula for the determination of squat:

$$Z = 2.4 * \frac{\Delta}{L_{pp}^2} * \frac{F_{nh}^2}{\sqrt{1 - F_{nh}^2}} \qquad \text{Eq. 2.2}$$

where,

Δ: volume de displacement [m^3],

L_{pp}: length of ship between perpendiculars [m],

F_{nh}: Froude number = $V/\sqrt{gh}$,

V: speed [m/s],

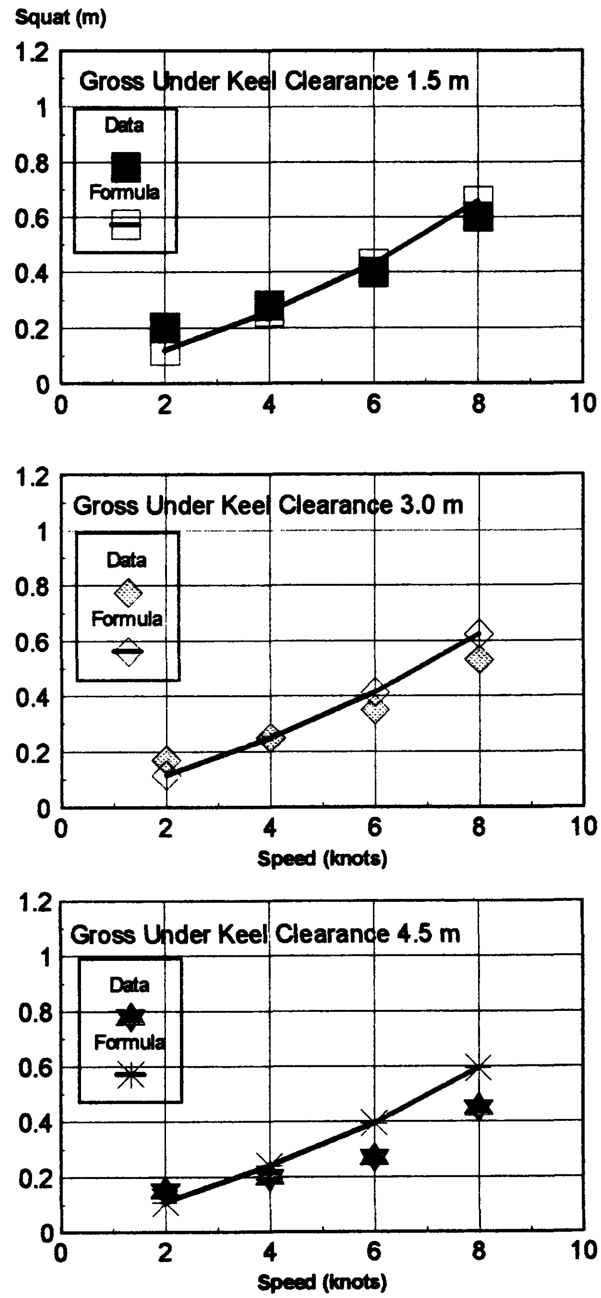
g: gravity [m/s^2] and

h: water depth [m]

As appreciated in figure 2.2, equation 2.2 agrees with the data from the National Hydraulic Laboratory. The figure shows the little effect in bow squat for velocities of less than six knots. Nevertheless, as velocity increases, the squat is more pronounced for low keel clearances than for higher ones.

Emphasis is placed in squat determination because it is the only factor in navigation depth definition directly related to ship transit. Since channel capacity depends on the ship's velocity and different velocities result in

Figure No. 2.2
EFFECT OF SPEED ON BOW SQUAT



different navigation depths, the squat will be considered when evaluation dredging requirements.

2.2 SAFETY IN NAVIGATION

Good engineers always try to maintain a good margin of security while keeping costs down. This sounds almost like an oxymoron for the better the security margin the higher the costs usually are. Keeping this in mind, a review of safety in navigation will be conducted. The review will have these basic objectives:

(a) to support the premise that a traffic management system can contribute to a reduction in ship collisions and

(b) to show that a basic assumption in navigation can be modified to optimize traffic flow with no loss in safety.

The first objective is easy to prove. A study made in Europe by Kemp (Ref. 13) tried to assess potential benefits of shore-based marine traffic systems. The study surveyed mariners and shipmaster to find the possible effect of different levels of traffic control: no traffic control, traffic control through a VHF reporting system with no radar surveillance and traffic control through VHF reporting system with full radar surveillance. The exercise concluded that collision casualties could be reduced 40% by using a VHF reporting system and 50% by the combined use of radar and a VHF reporting system; without a traffic control system no changes would occur on collision rates. The same paper concluded that a VHF reporting system alone would not affect stranding rates, while a combined system of VHF communication and radar surveillance would reduce strangling rates by 50%. Unfortunately, specifications for what information the radar surveillance was to provide are not included with the results.

The same study recorded collision data in European waters from 1978 to 1982. One interesting fact found is that only about 25% of all casualties occurred in restricted waters. Also, meeting or overtaking as reasons for collision outside ports accounted for about 70% of casualties while crossings accounted for the rest. Although the report does not mention the effect of traffic control in these percentages, its influence is likely to be low due to area covered by the study and the sparseness of such systems in operation even today.

Now its time to have a look at what causes a collision. As mentioned before, crossing causes about 30% of all collision in European waters. It is also very unusual to have crossing on a navigation channel. Subsequent analysis will exclude this type of collision. Bruun (Ref. 5) reports some data on marine casualties for Norwegian registered ships in the years 1985 and 1986. Although the data is limited, it does show "maneuvering and navigation" as a major reason for serious casualties: about 80%. Another interesting fact is that most grounding and collisions seemed to occur in clear weather with good visibility and calm and moderate wind conditions.

The same author reports data from the Elbe river, access to the port of Hamburg. This data is presented in table No. 2.2. Under the category "Maneuvering / Navigation" all causes related to insufficient navigation information or human errors have been included.

TABLE No. 2.2

CAUSES OF COLLISION AND GROUNDING CASUALTIES
IN THE ELBE IN THE YEAR 1970-1981

CASUALTY CAUSE	Grounding		Collisions	
	#	%	#	%
Ship's technical Failure	5	24.0	14	12.5
Maneuvering / Navigation	11	52.0	78	70.0
Other	5	24.0	20	17.5
TOTAL	**21**	**100.0**	**112**	**100.0**

The historical data presented supports the argument that a traffic management system can correct -or at least reduce- the number of collisions since most collisions are due to "maneuvering and navigation", therefore such system could conceivably improve safety.

The second objective of this review of safety is harder to prove. It has been claimed that a basic assumption in navigation can be modified to optimize traffic flow with no loss in safety. Stopping distance is that basic assumption. It is the minimum distance it takes a ship moving under its own power to come to a complete stop under emergency conditions. In channels and in channel capacity models it serves to define the separation distance between ships. The problem is that presumably the immediately preceding ship has stopped suddenly and therefore is a fixed obstacle. That is hardly the case in reality. A ship that experiences technical failures does not stop suddenly but maintains its motion for a longer stretch than a ship using its power to come to a complete stop. No ship will come to a complete sudden stop unless it finds a massive obstacle on its path and a navigation channel is not supposed to have any such obstacles. The only instance this might happen could be if, in a two-way channel, a ship loses control and blocks the opposing lane. Here, no amount of separation distance will help avoid a collision because separation distance refers to distance between vessel moving in the *same* direction.

Nevertheless, there is strong reason to consider stopping distance as equal to separation distance. First, a preceding ship might not be able to stop suddenly but it might slow considerably before the following ship notices and reacts. Therefore, for safety reason, stopping distance seems like the more reasonable alternative in defining separation distance. Now the key element is information. If the following ship is alerted immediately of the preceding ships maneuvers, it will have ample time to react and avoid a collision. Stopping distance, as a measure of separation, results in an excessive safety margin under this circumstance. The preceding argument is the base for separation distance proposals described next.

2.2.1 Stopping and separation distance

The idea of stopping distance is very important in channel capacity models. This distance defines the separation distance between ships and -as shown in subsequent chapters- it has important effects on the number of ships that can be served by a navigation channel. That is the reason behind the objectives presented under navigation and safety.

PIANC (Ref. 17) suggests the stopping distance should be no less than five times the length of the ship. It also recommends the next formula for vessels traveling at speeds exceeding 2.5 m/s (about five knots):

$$D = 4 * L * \left(\frac{V}{2.5}\right)^{.25} + L \qquad \text{Eq 2.3}$$

where,

D: stopping distance,

L: ship's length and

V: ship's velocity.

The preceding equation corresponds to a stoppage maneuver with engine half astern in deep water. Soo Lim (Ref. 20) for his channel capacity model modified the previous formula. He used a safety margin of 1.8 to come up with the following:

$$D = (.168 * V_m^{0.75} + 1.8) * L \qquad \text{Eq. 2.4}$$

where,

D: stopping distance,

V_m: ship's velocity and

L: ship's length.

Soo Lim offers no explanation for the usage of a safety margin of 1.8 but argues that his formulation is more effective in evaluating major factors in capacity such as fleet mix, channel length and vessel speed. He also implies that equation 2.4 was derived for deep waters, not the case in a dredged channel and therefore the need for an additional safety factor.

Bruun (Ref. 5) offers this equation for crash stopping distance in maneuvering speed:

$$D = \frac{m_o}{k} * \ln\left(1 + \frac{k u_o^2}{Tps\,(1 - t_p)}\right) + \frac{1}{2} u_o t_r \qquad \text{Eq. 2.5}$$

where,

D: stopping distance,

m_o: virtual mass of ship,

k: coefficient of ship hull resistance,

u_o: speed of ship before engine reversal,

T_{ps}: backing thrust of propeller

t_p: trust deduction coefficient of propeller and

t_r: time required for propeller stopping after engine reversed.

Kray (Ref. 14) lists the values tabulated in table No. 2.3 for vessels traveling at different speeds under favorable depth of water, weather and current conditions. Those values correspond to actual field data compiled by Shell Corporation.

According to Shell's information, a 206,000 DWT. ship moving at a speed between 14 and 18 knots requires about 17 ship lengths to come to a complete stop. The same ship moving at 12 knots requires some seven ship lengths: less than half the distance with a velocity decrease between 15 and 33%. Figure 2.3 shows this relation graphically. However, the same ratios do not hold for other ship sizes and velocities. For instance, a 65,000 DWT. vessel seems to require about 12 ship lengths to stop while a much larger vessel (200,000 DWT.) would need about seven ship lengths.

Another study by Delft (Ref. 6) for 160.000 DWT and 260.000 DWT vessels resulted in the values shown in table No. 2.4. This study was conducted using a model traveling on a canal five beams wide. The authors point out that the stopping distance ratio for both ship sizes was very close to their displacement ratio.

TABLE No. 2.3

EMERGENCY STOPPING DISTANCES AND TIME FOR COMPLETE STOP -SHELL

	Ship's Velocity		
SHIPS Fully Loaded	**14-18 knots Full Ahead**	**12 knots Half Ahead**	**4-8 1,100knots Slow Ahead**
65,000 DWT		2,620 m 10 min	
79,000 DWT			880 - 1,100 m 6.3 - 9 min
120,000 DWT	3,960 m 13 min		880 - 1,520 m 3.5 - 5 min
150,000 DWT	5,330 m 16 min		
206,000 DWT	5,640 m 21 min	2,190 m 10 min	1,310 m 9 min
300,000 DWT	4,330 m 24 min		

Note: data for 300,000 DWT vessel corresponds to an initial speed ahead of 15 knots.

Equations 2.3 to 2.5 result in different values for stopping distance. A comparison is graphed in figure No. 2.4 for different ship sizes. The comparison is not favorable. For larger ships (160.000 DWT.) empirical data from SHELL agrees with Soo Lim and the safety factors he proposes while Delft's data for a restricted waterway and low velocities tend to fall between Soo's and PIANC's equations. On the other hand, empirical data for a smaller ship (85.000 DWT.) agrees with PIANC's equation. Nonetheless, not enough data are available to make a reasonable case either way.

TABLE No. 2.4

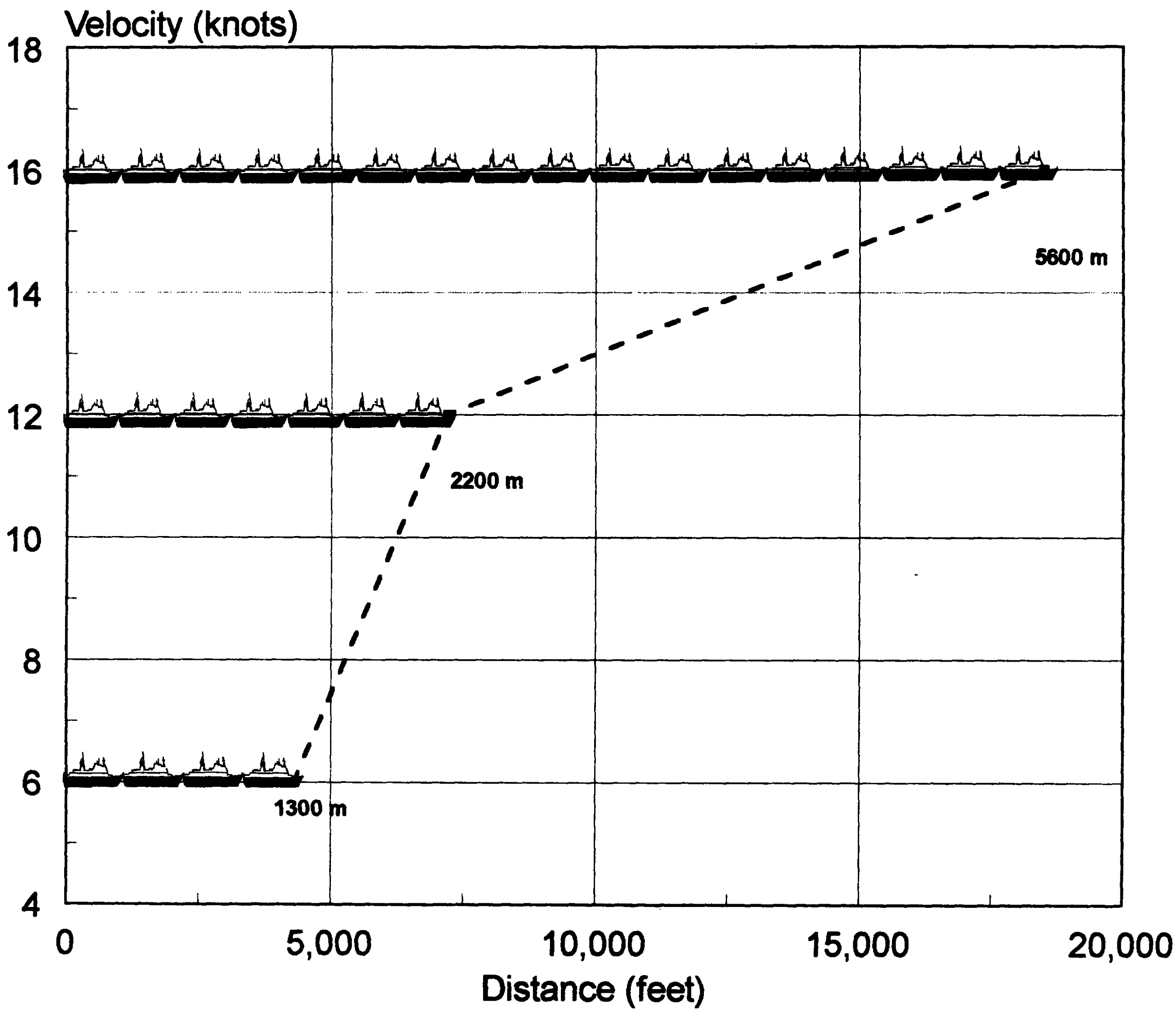
EMERGENCY STOPPING DISTANCES AND TIME FOR COMPLETE STOP -DELFT

	Ships's Velocity		
SHIPS	**6.5 knots**	**7.5 knots**	**8 knots**
160,000 DWT	2,250 m 11.2 min	2,950 m 12.6 min	3,300 m 13.2 min
260,000 DWT	2,750 m 13.7 min	3,800 m 16.2 min	4,350 m 17.4 min

Stopping distances have been obtained using the preceding formulas for three different ship sizes: 40.000, 85.000 and 160.000 DWT. in figure No. 2.5. Now assume those vessels are moving at speeds of 10, 8 and seven knots respectively, each on a different lane and each followed by another vessel. The possible combinations are tabulated on Table No. 2.5. That table shows the minimum distance each vessel would require to avoid a collision in an emergency assuming immediate response on both ships. The

Figure No. 2.3

EMERGENCY STOPPING DISTANCE
205,000 DWT VESSEL



first value tabulated results from using Soo's formula and the number in the parenthesis from PIANC's formula.

If the response is not immediate, but there is a time lag, the required separation distance would increase. Suppose now that a traffic management system can provide accurate information on another ship's position and velocity within a 5 min. interval. Then, the ship response would occur within 5 min and the separation distance needed would be as shown in table No. 2.6. This separation results from adding the distance traveled by the ship in a 5 min interval plus the minimum separation distance calculated assuming instant ship response.

TABLE No. 2.5

MINIMUM SEPARATION DISTANCE ASSUMING INSTANT SHIP RESPONSE (in meters)

Lead Ship			
Following Ship	**40.000 DWT**	**85.000 DTW**	**160.000 DTW**
40.000 DWT	0 (O)	0 (0)	0 (0)
85.000 DWT	210 (115)	0 (0)	0 (0)
160.000 DWT	320 (190)	125 (70)	0 (0)

A five minute response time is chosen based on Abdegail's paper on stopping distance (Ref. 1). This author says that human responses lag by 3 ± 2 min. at sea, thus the selection of a five minute response.

TABLE No. 2.6

MINIMUM SEPARATION DISTANCE ASSUMING A 5 MIN DELAY IN SHIP RESPONSE (in meters)

	Lead Ship		
Following Ship	**40.000 DWT**	**85.000 DWT**	**160.000 DWT**
40.000 DWT	1540 (1540)	1540 (1540)	1540 (1540)
85.000 DWT	1450 (1355)	1240 (1240)	1240 (1240)
160.000 DWT	1400 (1270)	1205 (1150)	1080 (1080)

A comparison between minimum separation distance obtained assuming a 5-min delay response and stopping distance results in the following observations:

(a) Using Soo's formulation for separation distance is similar to defining a reaction time of 10 min. In other words, it results in a separation distance that is about twice that obtained for a 5 min. delay in ship response.

(b) The equation proposed by PIANC for stopping distance agrees with the distance found assuming a 5 min. delay in response. PIANC's equation results in a distance that is at least 15% higher than using a 5 min. delay criterion (or equivalent to a 6 min. delay response). This is true even for extreme cases (not included in the graphs): a 160.000 DWT. ship traveling at 20 knots will require 3730 m to come to a complete stop (using PIANC's equation). If it is following a small (40.000 DWT.) ship traveling at five knots, it will require 3960 m. to stop using the 5 min. response strategy.

(c) No overtaking will occur during an emergency stop when a fast ship follows a slow ship due to the difference in stopping distance (The fastest ship will stop first).

The data presented suggests that Soo's factor of security for stopping distance might be excessive. Of course, this is true if the pertinent information is available to the navigator and the vessel is traveling in deep waters. The empirical data available supports this assessment for 85.000 DWT. ships, but not for the larger 160.000 DWT. vessels. Again, the empirical data is too scarce to help in defining stopping distance. Therefore, both formulations will be considered valid but under different circumstances. What this circumstances are, lead to three cases when evaluating capacity:

(a) A first case in which no additional information is available to the navigator: for this instance Soo's equation with its security factor seems the most reasonable course due to the uncertainties inherent to navigation (visibility, human error, etc.).

(b) A second case would assume the immediate availability of information but a 5 min. delayed response. Here, PIANC's equation will define separation distance because of its closeness to the 5 min. delay response criteria.

(c) A third case that presumes immediate availability of information and immediate response. Here, because separation distances are very short (see Table No. 2.5), two ship lengths will be used as separation distance.

One final observation. Shell's data includes the time required to reach a complete stop (Table No. 2.3). At low velocities (below 12 knots), a 5-min delay criteria would result in a reasonable separation distance since it takes different ships anything between 3.5 and 10 min. to come to a complete stop. This case would also support the assumption that the lead vessel would not become a fixed obstacle in an emergency. On the other hand, at larger velocities (over 12 knots) this does not hold true. There is quite a difference in experimental data for stopping time between a 120.000 DWT. vessel (13 min) and a 300.000 DWT. vessel (24 min). If the smaller ship tried to stop, it would become a fixed obstacle in about 13 min and this would violate the assumption about a ship becoming an obstacle. However, the important factor here is separation distance and even then the lead ship is not overtaken.

2.2.2 Ship's domain

The area around a vessel underway which most navigators of other vessel would avoid entering defines the ship domain. Separation distance between ships decides the longitudinal direction. Hydrodynamic interaction between ships and cross-section defines the lateral dimension. These dimensions define an ellipse that surrounds the ship in all directions.

Two zones will be defined within a ship domain. A first zone in which the longitudinal distance is the minimum required for a complete stop assuming information is available to the pilot. This area will be the danger zone. Any ship coming into this area is potentially dangerous. The second zone corresponds to the extended distance required when no information is available. Lateral distance remains constant because hydrodynamic interaction does not change. Note that a ship's domain will vary with fleet mix and traveling velocity according to the formulations discussed in the previous point. Also, since the longitudinal dimension is based on stopping distance, it will be measured from the bow of the vessel (See figure No. 2.6).

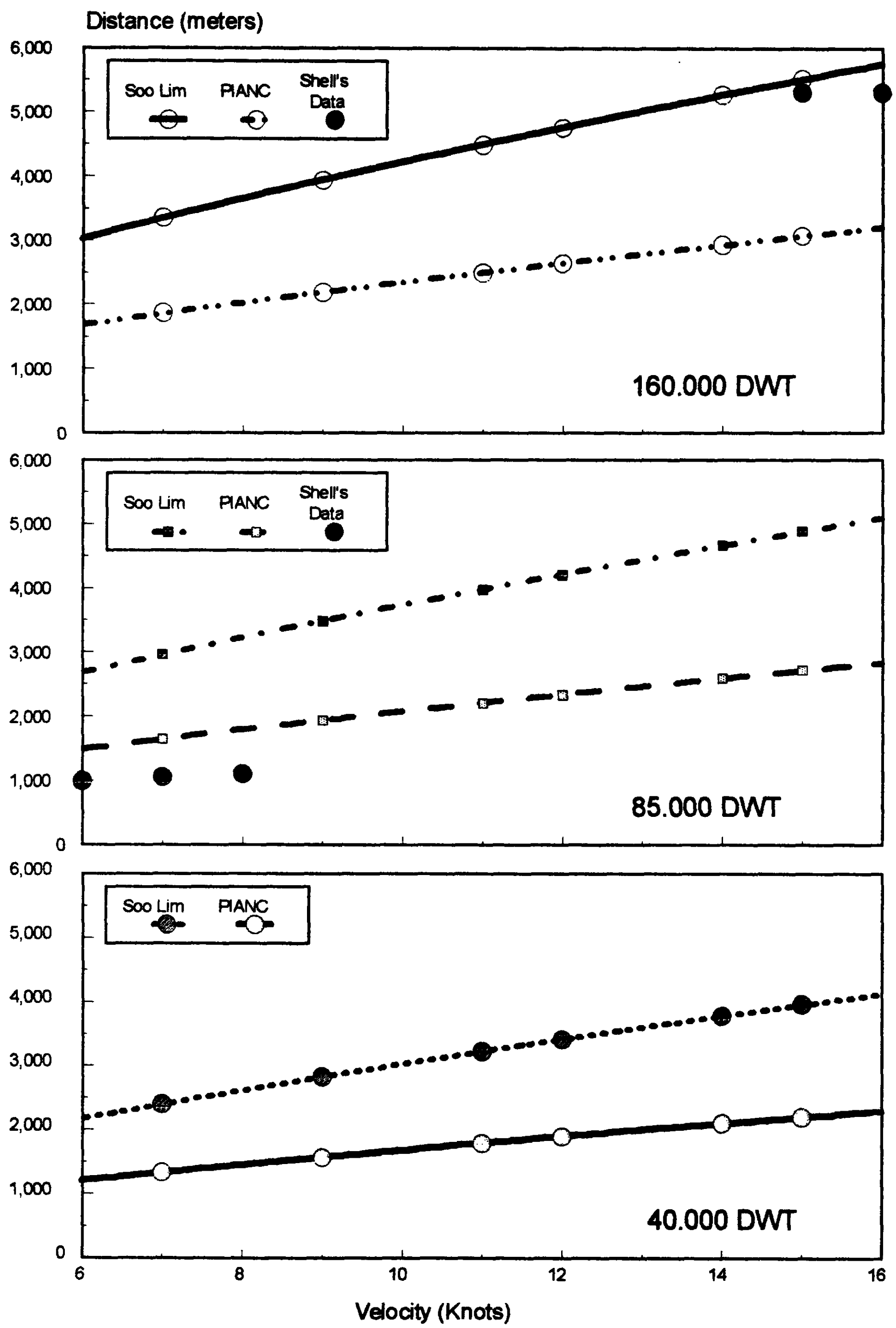
The traffic management system requires this differentiation. If a lead ship enters the danger zone of a ship it is already too close and the navigator should notice this and act accordingly. While the lead ship is within the safety zone, the navigator should monitor the lead ship performance closely but no action should be required. The effect of ship domain on traffic control will be discussed later.

2.3 SUMMARY

Traffic control can result on improvement in a ship's maneuverability and therefore can serve to redefine channel's dimensions. Depending on the level of traffic control, a channel's width could be reduced from five to four beams of the design ship for a one way channel. For two way channels, the width could be decreased from eight to six design vessel beams. This will not result in unsafe operations if a vessel's navigator has continuous information on its position. Hence, a reduction in channel width will be conditioned by the presence of a navigation control system.

Figure No. 2.4

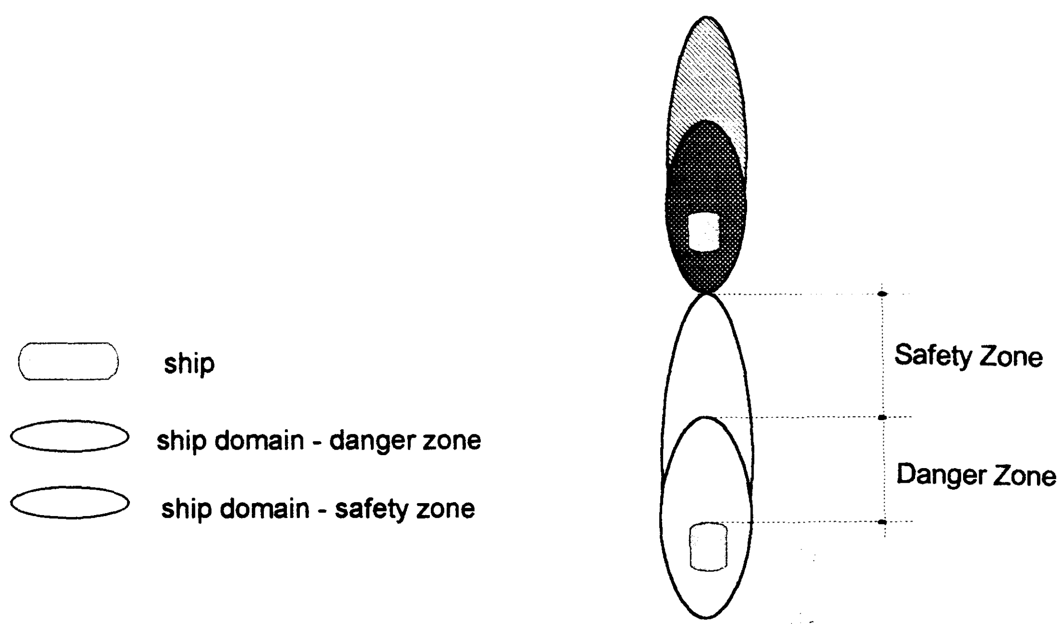
STOPPING DISTANCE COMPARASION



A traffic control system can also help reduce the inter-ship separation distance. Defining separation distance as equal to stopping distance is not necessarily the best and safest strategy. It has been shown that it is highly unlikely that a ship will find a fixed obstacle on a navigation channel. Therefore, the separation distance within the channel can be defined by reaction time. Using this criterion, separation distance can be reduced. In order for this to work the navigator must have information of other vessels' positions: information that can be provided by a traffic control system.

Figure No. 2.5

SHIP DOMAIN



CHAPTER No. 3

3. TRAFFIC OPERATIONS AND MANAGEMENT

Up to this point, much paper has been dedicated to define the navigation channel in terms of design and how it affects capacity. This chapter will be directed toward actual traffic simulation and its applicability to improve a channel's capacity. A mathematical model developed by Soo Lim (Ref. 20) will help in highlighting important factors to be considered when planning traffic control.

Based on his results, the factors to change during simulation to find their influence on capacity are: Queuing mode, ship velocity, fleet mix, separation distance, channel length, number of lanes, velocity distribution and arrival rate. There is an almost infinite set of different combinations of the previous factors -at least for practical purposes. Consequently, the goal will be to find general tendencies in his results. These tendencies will hopefully suggest a best strategy for traffic control.

The model by Soo Lim served as the basis for the model proposed in this paper. The new model allows for simulation when actual ship arrival information is readily available or for modeling when new trends or conditions require changes of strategy. It was written in VISUAL BASIC, a programming language easily accessible and with a Windows like interface to allow for ease of use. The basic premises of Soo Lim's model are maintained while incorporating a more flexible and user friendly interface.

3.1 A FEW BASIC CONCEPTS

Before a traffic model can be discussed, some ideas related to capacity and queuing theory must be reviewed.

(a) *Service time*

In Queuing Theory, service time refers to the length of time a customer spends in the serving facility. Queuing theory prohibits the next customer to use the facility during that period. Then, the service time is the time during which an incoming ship cannot use the channel or, in other words, the minimum time of separation. Under this conditions, service time is dependent on the immediate following vessel but independent of the immediate preceding one. The service time will be calculated using the following formula when the maximum separation distance (as outlined in point 2.2.1) and one speed case (all vessels travel at same speed) are considered:

$$E[S]_o = \frac{E[L]}{V_i} * (.168 * V_i^{.75} + 2.8) \qquad \text{Eq. 3.1}$$

where,

$E[S]_o$: is the expected service time for a one speed case,

E[L]: is the weighted average of vessel length and

Vi: is the ship's velocity

The service time for the minimum separation distance and one speed cases is calculated using the next equation.

$$E[S]_o = \frac{E[L]}{V_i} * (.094 * V_i^{.75} + 2.0) \qquad \text{Eq. 3.2}$$

where,

$E[S]_o$: is the expected service time for a one speed case,

E[L]: is the weighted average of vessel length and

Vi: is the ship's velocity

For the multiple speed case (vessels travel at their own allowed speed), the relative vessel speed decides the expected service time. Therefore, the following equation must be used in those cases:

$$E[S] = E[S]_o + \sum\sum (LC-D) * MAX[\frac{1}{Vi} - \frac{1}{Vj}, 0] * P_i * P_j \qquad \text{Eq. 3.3}$$

where,

E[S] is the expected service rate for the multiple speed case,

$E[S]_o$: is the expected service time for a one speed case,

LC: is the length of channel,

D: is the separation distance calculated according to equations 2.3 or 2.4 for maximum and minimum values respectively,

Vi, Vj: are ship's velocities and

Pi, Pj: relative proportion of ship i and ship j in fleet mix.

(b) *Waiting time*

The waiting time will be defined as the time elapsed between ship arrival and channel entrance. This time will vary with the arrival rate, queuing routine, fleet mix and other factors. It will serve as an index of capacity.

(c) *Waiting cost*

A waiting ship is a money losing ship so the cost of waiting will be calculated. This factor represents the cost associated to the total waiting time: that spent in queue and on the ready spot. It will also serve as an index of capacity.

(e) *Arrival rate (λ)*

The frequency of ship arrival defines the arrival rate. It is a measure of a channel's capacity to serve ships. The maximum inter-ship arrival rate (λ_{max}) is the shortest interval in which ships can arrive to the channel entrance without causing an infinite queue. As the length of the channel and the separation distance growths, the maximum interarrival rate diminishes. To find out the maximum inter-ship arrival rate, the following expression will be used:

$$\lambda_{max} = \frac{1}{E[S]} \qquad \text{Eq. 3.4}$$

where,

λ: arrival rate and

E[S]: the expected value of the service time.

(f) *Total waiting cost*

Total waiting cost is the summation of waiting costs of all vessels during the planning period.

(g) *System time*

The time a ship waits to enter the channel plus the time it takes a vessel to traverse the channel make up the system time. This value will serve as an index for cost estimation and capacity.

3.2 QUEUING (OPERATIONAL) MODE

Three basic queuing models apply for traffic control in a navigation channel. These models are:

(a) First come, first served (FCFS),
(b) Faster vessel first (FAST) and
(c) Higher-cost vessel first (COST).

The model First come, First Served (FCFS) is the universal one. This model is the one used when no strategy is in place. As its name implies, it allows in whatever vessel gets to the entrance to the channel first regardless of waiting cost, ship size or velocity.

A second model "Faster vessel first" (FAST) arranges ship entrance to the channel according to their velocity. The basis for this strategy is the fact that the channel serves more vessels if the fastest ship is allowed to go first. It is an effective model from number of ships point of view. Yet, it has the draw back that the fastest ship is not necessarily the more expensive one and usually results in higher global waiting costs.

Finally, the third model is the COST model or highest-cost vessel first. This strategy allows the ship with the highest waiting cost to go first into the channel. This results, of course, in lower global waiting costs. It does not, however, permit as many vessels in as the FAST model allows for the same period.

Additional to these three basic models, a fourth variant will be considered: Convoys. This last model operates under either of the previous modes (FCFS, FAST or COST) to make the decision on which the vessel will have priority once a previous convoy has been filled and then allows for groups of ships of the same *type* to be given priority.

The benefits to be derived from the use of a given strategy will depend on the number of traffic lanes (one or two), the final objective of the user and the traffic level. In low traffic situations (less than 70% capacity), there is very little to gain from the use of COST or FAST when compared to FCFS in factors such as global cost or waiting time whatever the number of lanes although there is a slight improvement in global cost when the COST strategy is used or a slight improvement in global waiting times when the FAST strategy is in place.

Nevertheless, the relative benefits increase with traffic level. For instance, for an even fleet mix traveling on a two-way channel global waiting is reduced by about 20% if a COST strategy is used. When a FAST strategy is in place the reduction is about 40% (compared to FCFS strategy). The situation is reversed -for COST and FAST strategies- but similar when global costs are analyzed.

When convoys are used and traffic is high, the improvements are even more marked. First the service time improves, in effect increasing the capacity of the channel (This point will be discussed in more detain in point 3.6.1). Then the relative benefits of each strategy also improve by different but important margins. These margins vary with the length of the convoy. Depending on the channel's conditions and traffic, different convoys lengths result in maximum benefits. Here, improvements obtained for a certain convoy length and queuing strategy does not necessarily imply the same optimum length for another queuing strategy.

3.3 VELOCITY DISTRIBUTION AND SEPARATION DISTANCE

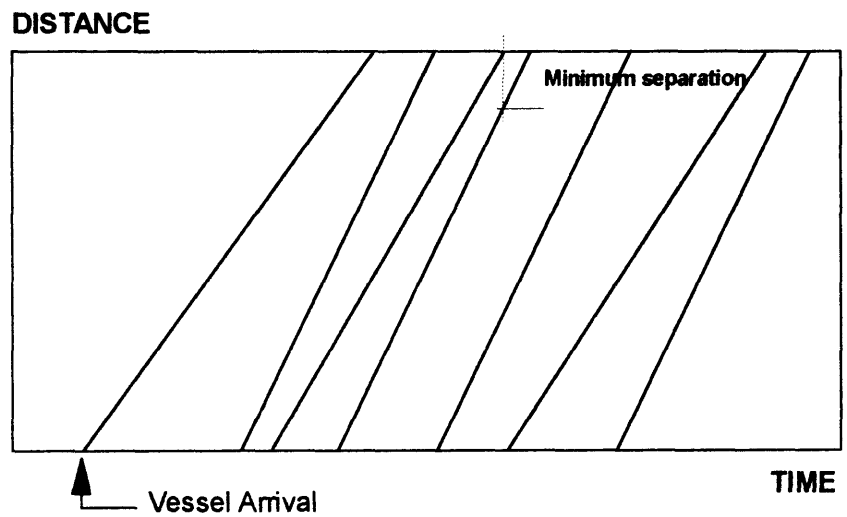
Soo Lim found that velocity distribution within a channel has a pronounced effect in capacity. Velocity not only decides the time it takes a vessel to traverse the channel, but it is also the base for the definition of separation distance and minimum time separation as outlined in chapter 2.

He evaluated two velocity modes: (a) one-speed and (b) multiple speeds. The one-speed mode assumes that a group of ships in a channel moves at the speed of the slowest vessel (See figure 3.1-a). The multi-speed mode allows the fastest ship to travel at its regular speeds but always maintaining the minimum separation distance as defined by eq. 2.4 (figure 3.1-b). Both velocity modes work for FCFS (first come, first served) and COST (high-cost vessel first) strategies. However, there is no sense in using a one-speed mode for a FAST (fastest vessel first) strategy or for convoys.

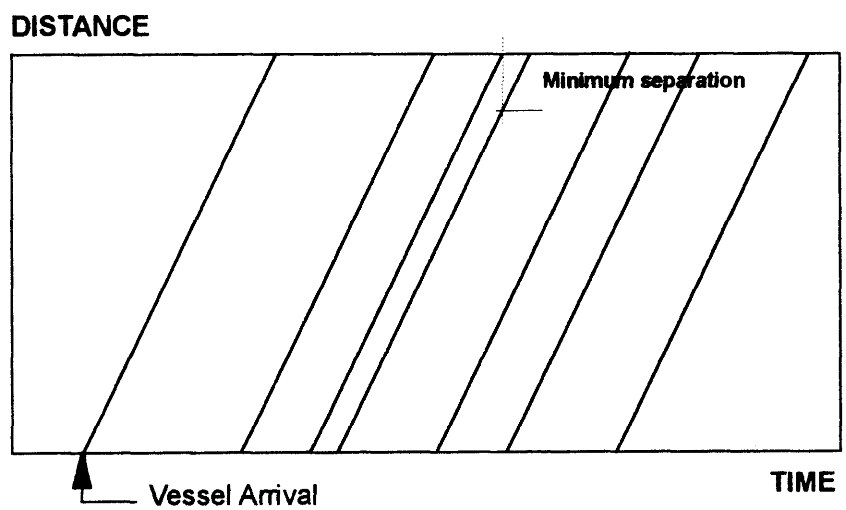
Figure No. 3.1

SPEED MODE COMPARISON

MULTI-SPEED MODE



ONE-SPEED MODE



The choice of velocity mode modifies the separation distance. In equation 2.4, the velocity is that of the slowest vessel when using one-speed mode. In multi-speed mode, the velocity to use in the equation is that of the vessel.

3.4 TRAFFIC SIMULATION MODEL

The traffic simulation model used as a base for the model presented in this paper was developed by Soo Lim (Ref. 20) using the simulation language SLAM II (*Simulation Language for Alternative Modeling*). This language combines network, discrete event, and continuous modeling capabilities. It provides network symbols for building graphical models that are easily translated into statements for direct computer processing. It also contains subprograms that support both discrete event and continuous model developments.

The basic simulation model divides the channel in three operation stages: waiting pool, ready spot and channel. Soo describes the stages the following way: "A vessel arrives at the waiting pool ... and joins the waiting queue. When the vessel which occupies the ready-spot enters the channel, the highest priority vessel in the waiting pool fills the ready-spot which has only one space. The priority of each vessel in the waiting pool is determined by the predetermined queue discipline and updated when a vessel in the ready-spot enters the channel. The vessel occupying the ready-spot in the next vessel entering the channel regardless of its priority. The vessel in the ready-spot enters the channel if the immediate preceding vessel in the channel is located far enough from the entrance of the channel." The model is then a translation of the fore mentioned process in mathematical terms.

Soo's model has a few drawbacks. It was written so a computer operator with basic queuing theory knowledge could make few changes and get results. The key here is the basic programming and queuing theory knowledge. The interface is not user-friendly and requires some time to master. Consequently, the new model changes this and allows for greater flexibility in choices and results. It can be readily modified to fit any navigation channel when a few basic statistics are known. It incorporates a few features to make the basic model easy to use while providing useful information not previously available. The model also allows for the simulation of actual ship arrival data and could provide a day to day program for greater efficiency in traffic management. In short, the philosophy behind the model produced in this paper is to develop a tool readily available for traffic control and management which is flexible and provides all required information.

3.4.1 Assumptions

The current model allows for modeling and simulation of data. The assumptions made for each different mode are:

(a) Poisson arrivals

Research by Plumlee (Ref. 20) suggests that ships arrive at public seaports following a Poisson distribution. His study shows agreement between predicted and actual distribution ranging from 81.5 to

98.5%. Due to this results, the capacity model assumes vessels arrive in a Poisson manner when in modeling mode. This means that ships arrive independently and that the inter-arrival time between successive ships is exponentially distributed.

(b) There is enough waiting space

Enough space is available at the entrance and exit of the channel to hold an infinite number of ships and the vessel exiting the channel does not block traffic. No limit in the number of vessels waiting is set in the mode; nevertheless, it will advise the user when an infinite queue is about to be generated.

(c) There is no time delay in changing stages

When changing from waiting pool to ready-spot to channel, the transition is instantaneous. The arriving time at the waiting pool and the entering time to the ready-spot is the same if the ready-spot is empty.

(d) The velocity is constant

A vessel maintains a constant velocity while traversing the channel. This does not mean all ships necessarily maintain the same velocity. Each vessel can travel at its own speed but that speed remains the same all along the channel.

(e) The time separation is deterministic

(f) Fixed fleet mix

There are only three possible vessel types allowed. When modeling, it will be assumed that fleet mixes do not change during the time horizon of the model.

(g) Vessel data is available

When simulating, all data required by the traffic controller is assumed to be available before the vessel's arrival so the controller can prepare a schedule. The data should include size, speed and expected time of arrival (ETA) of the incoming ship.

(h) Traffic flow is continuous

A channel can operate 24 hrs a day, 365 days a year. No accident will impede traffic nor will bad weather during the time horizon of the simulation.

(i) No passing zones exist

Previous assumptions are those suggested by Soo's with a few changes. Separation distance between vessels will be considered a variable, as will the speed mode (discussed latter on).

3.4.2. Description

The model is written in Visual Basic, a modular programming language which allows for a Windows-like interface. It has a main program with the basic calculation routines -that have the equations described previously- and several sub-programs reached through "forms" that handle data input and output (See Appendix A for programming code and Appendix B for user's manual). A general description is given in this point, a more detailed description is given in Appendix B.

There are three basic areas in the model that the user can modify. A first area contains the basic fleet and channel characteristics. The model allows for three different vessel types. Those ships are defined by DWT, length, beam, speed and waiting costs. Those types will serve as basis for the model. The channel length and the number of lanes must also be specified in this area.

On a second area, the user selects arrivals. He can generate random arrivals with a Poisson distribution or input actual vessels arrivals. When the user decides to generate arrivals, he must specify the maximum inter-ship arrival rate, the percentage of vessels of each type in the fleet mix and the time horizon of the incoming modeling action. This data can be saved on a file to be viewed later. The file contains arrival time and date, vessel type and priority. When the model generates arrivals, all vessels are assigned the same priority. On the same screen, the user can change the ship arrival data when the real time data is to be used to simulate. The user must specify the arrival time and date, the vessel type and whether the vessel will have priority over other vessels in the waiting pool. The program will automatically reorder the vessel data base to fit this vessel in a FCFS list saved on a separate file.

The third area controls to the actual operation. The output devices and parameters to be considered when running the model are defined in this area. First, the user must specify whether he wishes to simulate (with real time data) or model (with data generated by the program). The model will use a different file/data base concurring with this selection.

Before the user starts processing the data, he must set up the speed mode, separation distance, convoy length and queuing strategy. Different combinations might be specified simultaneously. For instance, if a user chooses FCFS, COST, multi-speed and one-speed mode for maximum separation distance, the results will reflect all combinations of the mentioned factors. Once the user has shown a set of parameters, they will be saved and will serve as default until a new set is defined.

The results of the simulation or model will be sent to the device selected by the user: printer, screen or file. The model gives an output for each vessel type and for the whole operation, consisting of the following indexes:

a. Service Time
b. Waiting Cost
c. Waiting Time
d. Throughput per unit time

The model also selects a *best* solution. This is specified by the user on this area. The criteria for best solution selection are:

a. Lowest overall cost
b. Lowest waiting time
c. Largest throughput per unit time

The program can select this optimum solution for single vessels or for different convoy lengths when the user so specifies.

It tries to incorporate all the elements described before (speed mode, operational strategy, channel and fleet characteristics, separation distance) so a manager has real time information for channel management. The idea was to introduce all the parameters a user might want to use to decide the best strategy for channel operations. Those parameters can be taken in separate or together when possible.

3.5 MODELING RESULTS

The capacity of the channel will be measured through five parameters: service rates, total throughput per unit time, global waiting and global waiting cost. The objective at this point is to guide the user on the choices the model offers to improve efficiency as well as the benefits of a navigation control system. To that end, the effect of different variables in model results is discussed. For comparison purposes, the service rate and total throughput per unit time are used. Note that the service rate defines the number of ships that can be served by the channel under a given mode. As described before, it depends on speed, separation distance and fleet mix. Therefore, each combination of fleet mix, separation distance and speed mode results on a different service rate as shown in table No. 3.1.

TABLE No. 3.1

DESIGN VESSELS

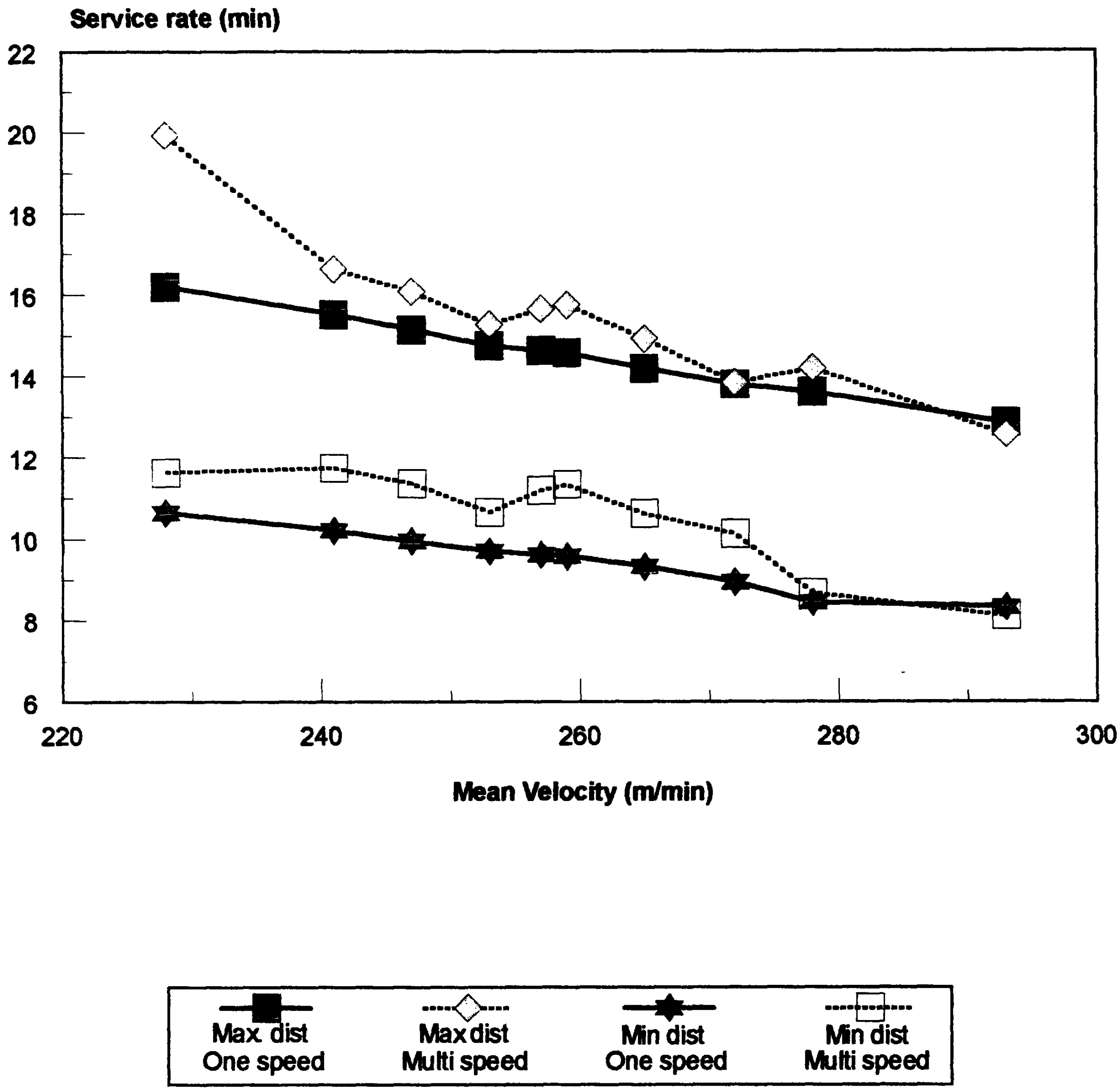
	TYPE 1	TYPE 2	TYPE 3
Dead Weight [ton]	160000	85000	40000
Speed [m/min]	216.1	246.9	308.7
Length [m]	297	263	213
Cost [$/day]	30240	20160	14400

Soo compares arrival rates -defined as one over the expected service rate- between different alternatives as a percentage of the maximum arrival rate for that alternative. This comparison basis can be misleading. For instance, when the service rates are different and the comparison criterion is a percentage of the maximum capacity, there is little difference in global waiting; however, when the ship inter-arrival rate is used, the difference is marked. This is because the *shape* (as defined by the relative values for the same percentages of usage) is the same for both speed modes, but the maximum capacity values differ.

There is, however, another good reason to use service rate as comparison criteria. For all the cases evaluated, despite queuing mode, the service rate is a direct indicator of capacity. The lower the service rate the lower waiting time, global waiting cost and global cost are. So in this sense, the service rate can, and will, be used as a "quick and dirty" indicator of capacity.

Figure No. 3.2

CHANGES IN SERVICE RATES DUE TO SPEED MODE AND SEPARATION DISTANCE



For comparison purposes, the model is run for 2.5, 5 and 10 NM long, two-way navigation channel. The figures shown in Appendix C show the basic results obtained using the model under the different modes discussed previously. The data presented there is the basis for the following analysis. Also, the fleet mix is varied but for only three vessels with the following characteristics:

3.5.1 Service Rates

The model results show no significant difference between the expected maximum service rate calculated using equations 3.1 to 3.3 -and a first come first serve strategy- and COST and FAST strategies for the one speed mode. When multi-speed mode is allowed, there is a slight improvement in service time when the FAST strategy is in place. No differences were detected when using either a COST strategy or when allowing for convoys. In short, the use of a different queuing strategy does not change the service rate. This is because queuing strategy basically affects the order in which vessels enter the channel (the server). It does not affect the separation distance or the speed mode -factors that do alter the service rate. This fact is shown in table 3.2.

The speed strategy that results in the lowest service rate is the one speed mode for channels longer than 4 NM for the fleet mixes evaluated; otherwise, a multi-speed strategy is better. When the fleet mix changes, the behavior is similar except for fleet mixes with faster speeds.

TABLE No. 3.2

SERVICE TIME COMPARISON (in minutes)
FCFS Queuing Strategy

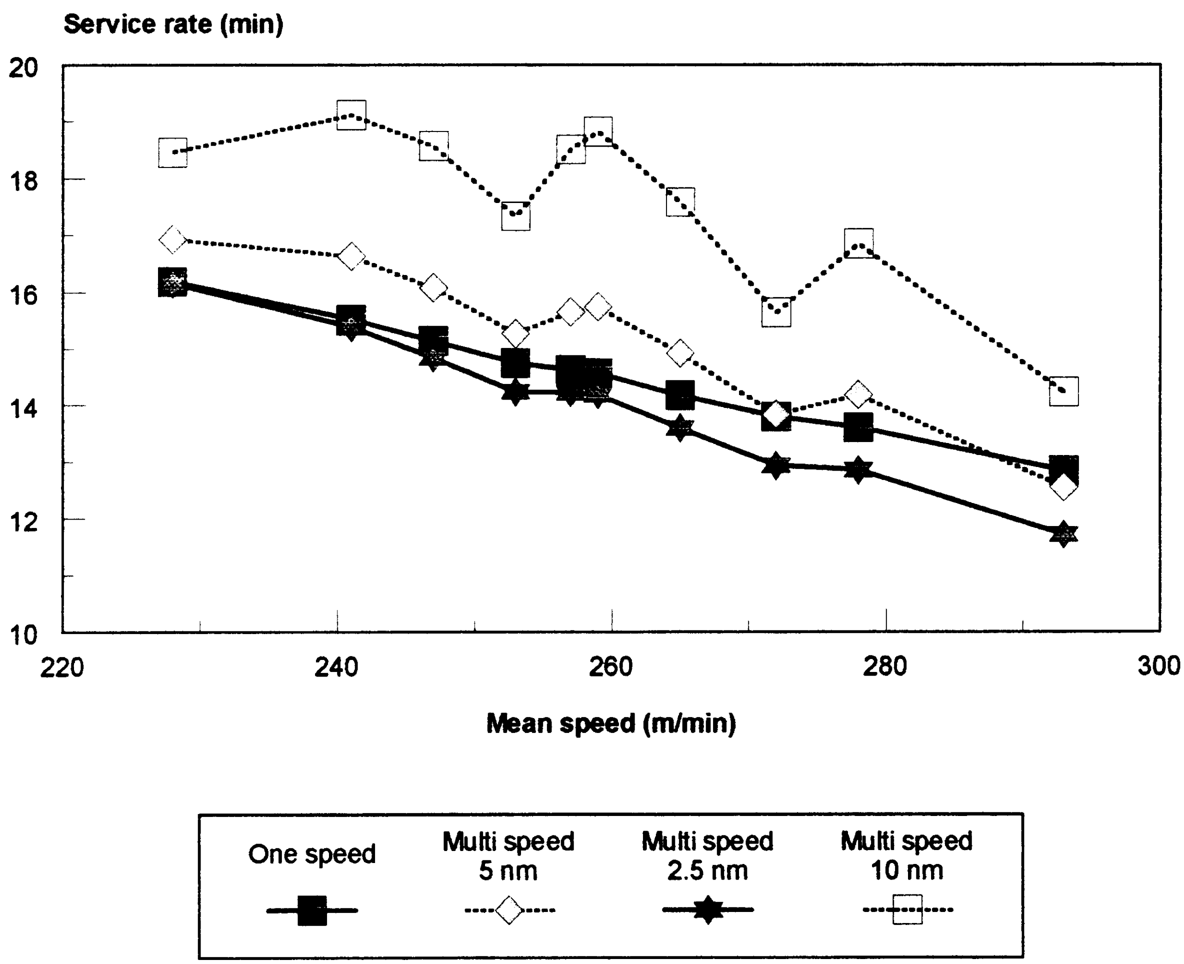
SPEED MODE	FLEET MIX	MAXIMUM SEPARATION EQUATION	MAXIMUM SEPARATION MODEL	MINIMUM SEPARATION EQUATION	MINIMUM SEPARATION MODEL
One	811	16.2	16.2	10.6	9.6
	333	14.6	14.6	9.6	8.7
	118	12.9	12.9	8.3	7.6
Multi (2.5 nm)	811	16.2	15.5	10.9	9.9
	333	14.2	13.6	9.8	9
	118	11.7	11.3	7.5	7.3
Multi (5 NM)	811	16.9	16.2	11.6	10.7
	333	15.7	14.9	11.2	10.4
	118	12.6	12.1	8.1	8.2
Multi (10 NM)	811	18.5	19.3	13.2	12.2
	333	18.5	19.3	14	13.3
	118	14.2	15.1	9.3	9.9

As the channel length increases so does the multi-speed mode service time. Also, increments in the relative proportion of type three (fastest) vessels are followed by reductions in service time as might be expected. Figure 3.2 shows these tendencies for different channel lengths. It is important to point out that these tendencies correspond to the vessels previously described. Changes in fleet mixes and velocities result in different behavior. Of course, the difference in service times between one and multi-speed modes is lower as the relative vessel speeds get closer.

Figure No. 3.3

CHANGES IN SERVICE RATES DUE TO CHANNEL LENGTH

MAX. DISTANCE SEPARATION



Separation distance is by far the factor with the largest effect on service time. As figure 3.3 shows, service times decrease by at least 40% if separation distance is kept at a minimum as opposed to maximum separation. This is true despite speed strategy and channel length. If for example the channel length is increased to 10 NM, the difference in service times due to separation distance is about 38% while the same difference for a 2.5 NM channel the reduction amounts to about 41%.

The data analyzed shows that:

(a) Queuing routine does not affect service rates.

(b) The best speed mode routine depends on the channel length and the fleet mix. For the examples evaluated, one speed mode is better than multi speed for longer channels, otherwise for shorter ones.

(c) The separation distance is a very important factor in service time definition. A lower separation distance always results in shorter service times.

3.5.2 Global waiting and Global cost

The two speed modes and two separation distances discussed before result in four cases: Omin, Mmin, Omax y Mmax. Omin shows one-speed mode with minimum distance (minimum distance as described in chapter No. 2) while Mmin indicates multi-speed with minimum distance. Omax indicates one-speed mode and maximum distance separation and Mmax defines multi-speed mode with maximum distance.

Figures 3.4 to 3.9 show the results obtained using the model for global waiting and global cost with a FCFS strategy on different fleet mixes and

Figure No. 3.4

EFFECT OF SPEED MODE ON GLOBAL WAITING
FCFS ROUTINE 2-WAY, 2.5 NM

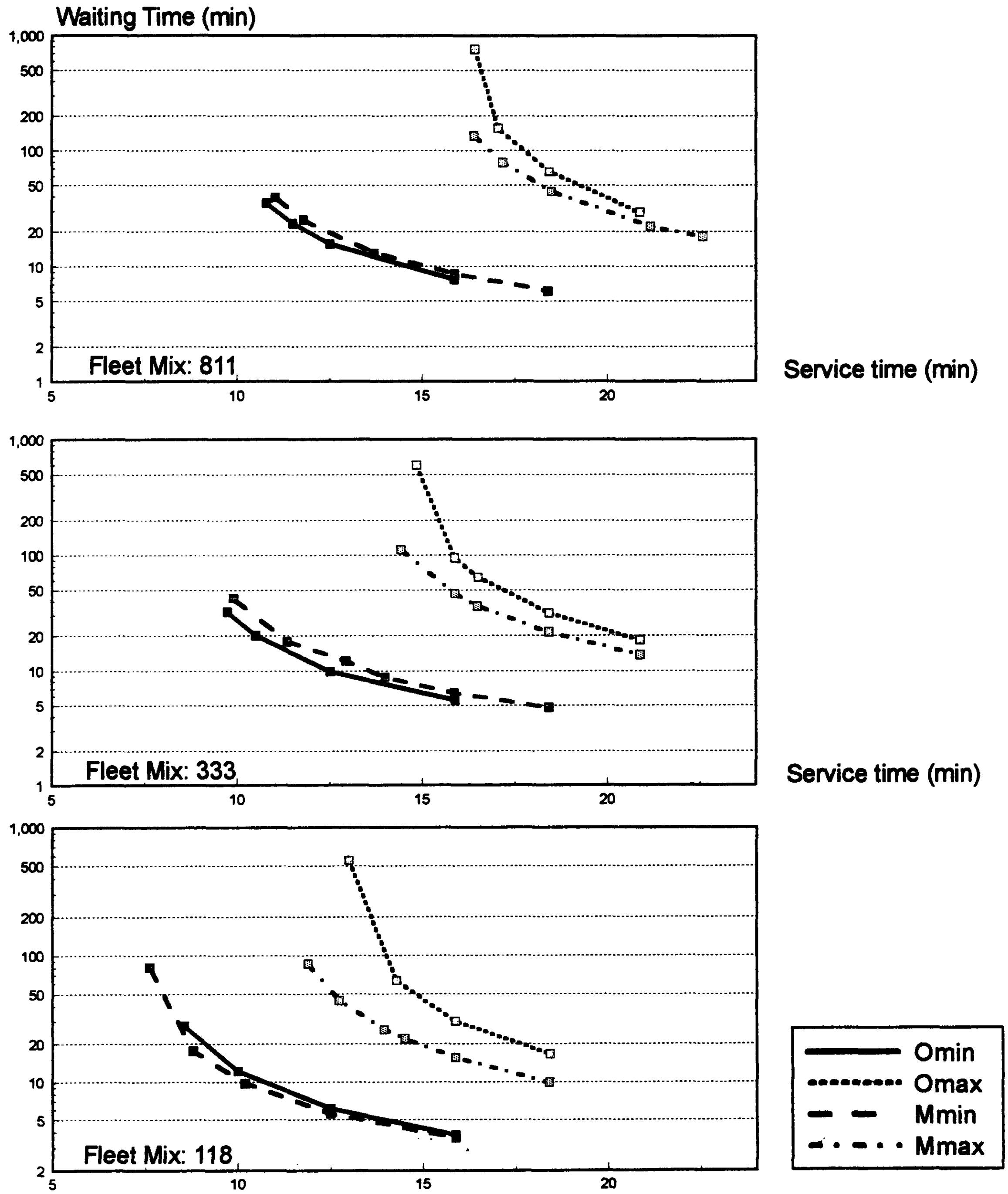


Figure No. 3.5

EFFECT OF SPEED MODE ON GLOBAL WAITING

FCFS ROUTINE 2-WAY, 5 NM

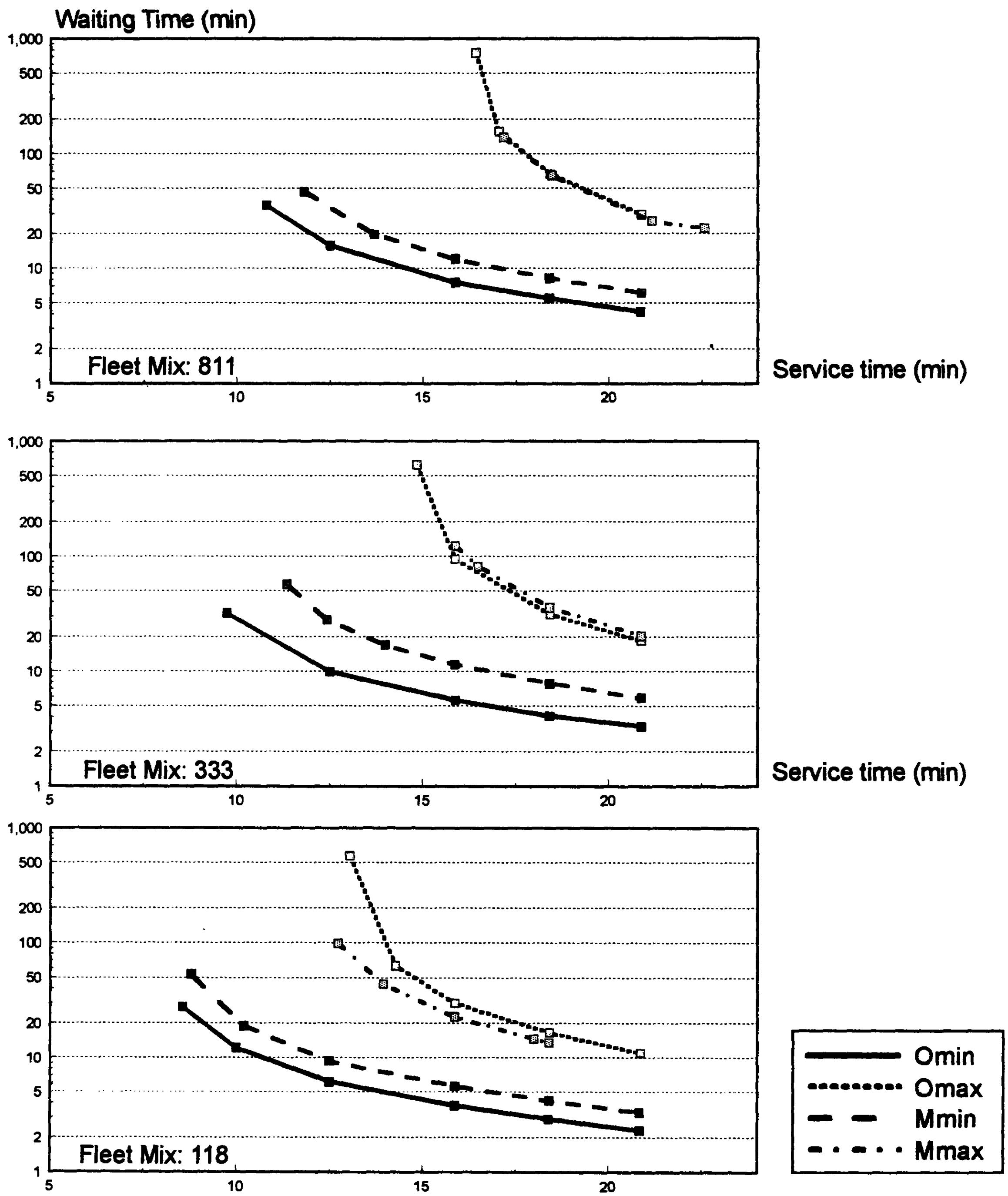


Figure No. 3.6

EFFECT OF SPEED MODE ON GLOBAL WAITING

FCFS ROUTINE 2-WAY, 10 NM

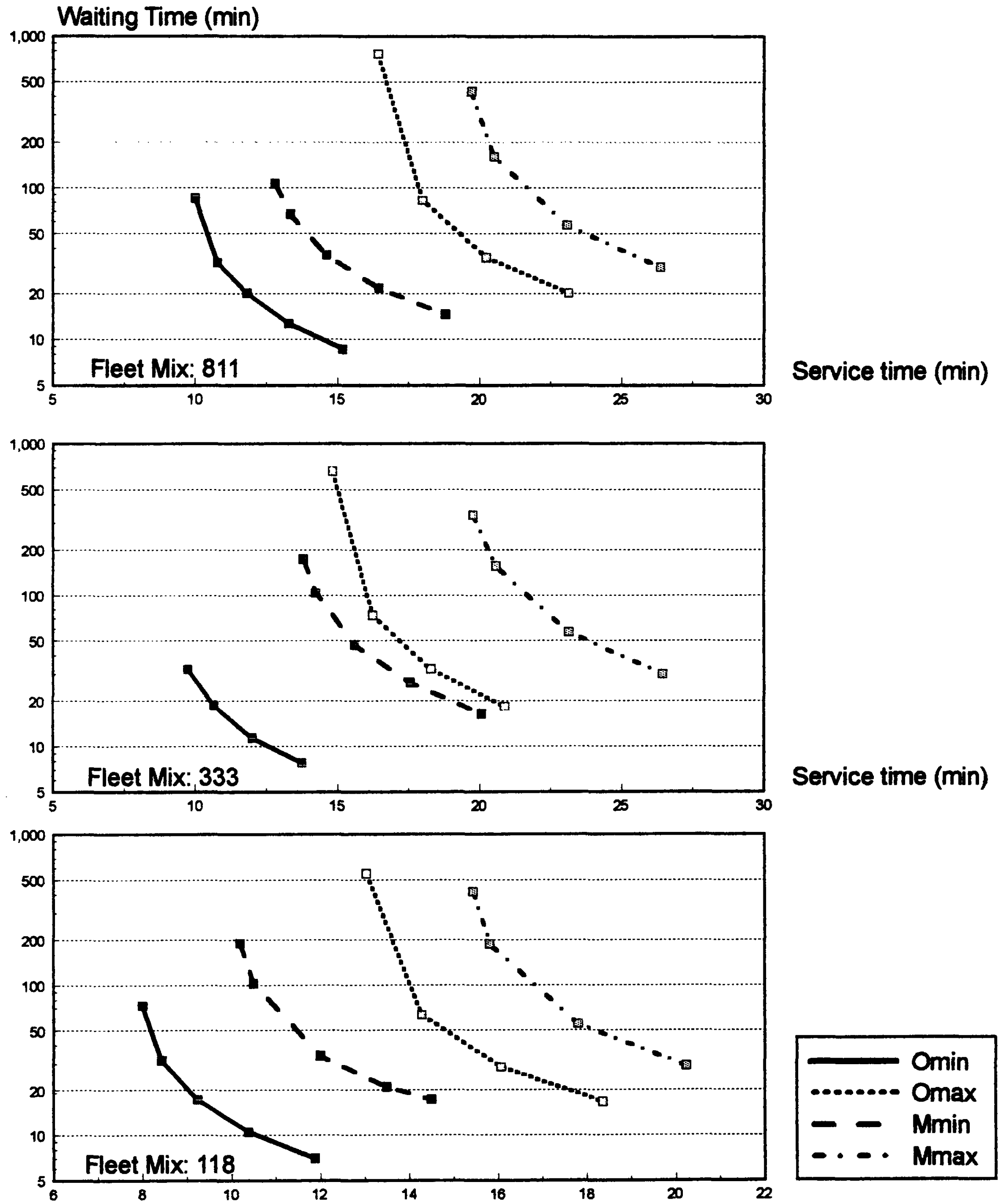


Figure No. 3.7

EFFECT OF SPEED MODE ON GLOBAL COST

FCFS ROUTINE 2-WAY, 2.5 NM

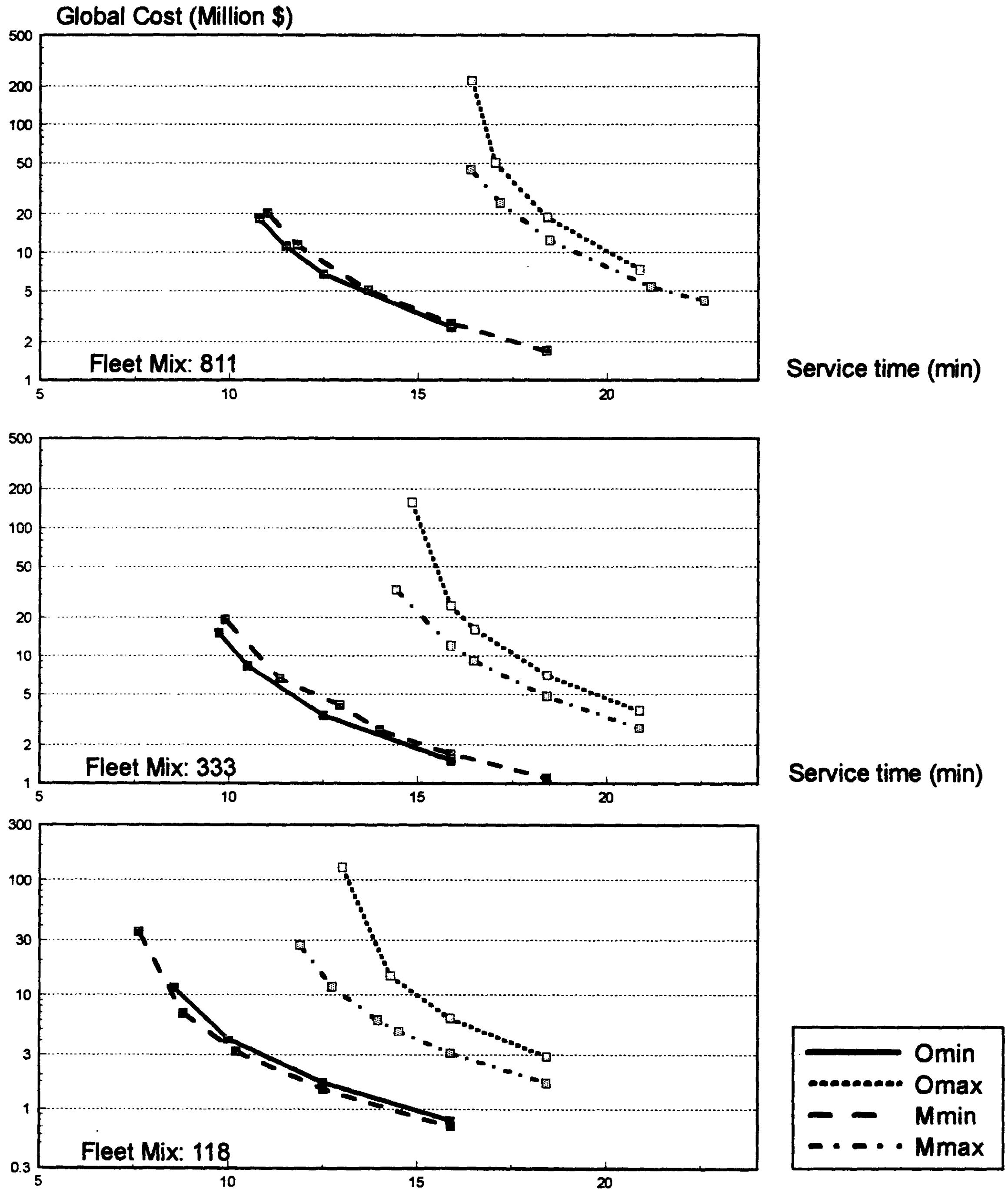


Figure No. 3.8

EFFECT OF SPEED MODE ON GLOBAL COST

FCFS ROUTINE 2 WAY, 5 NM

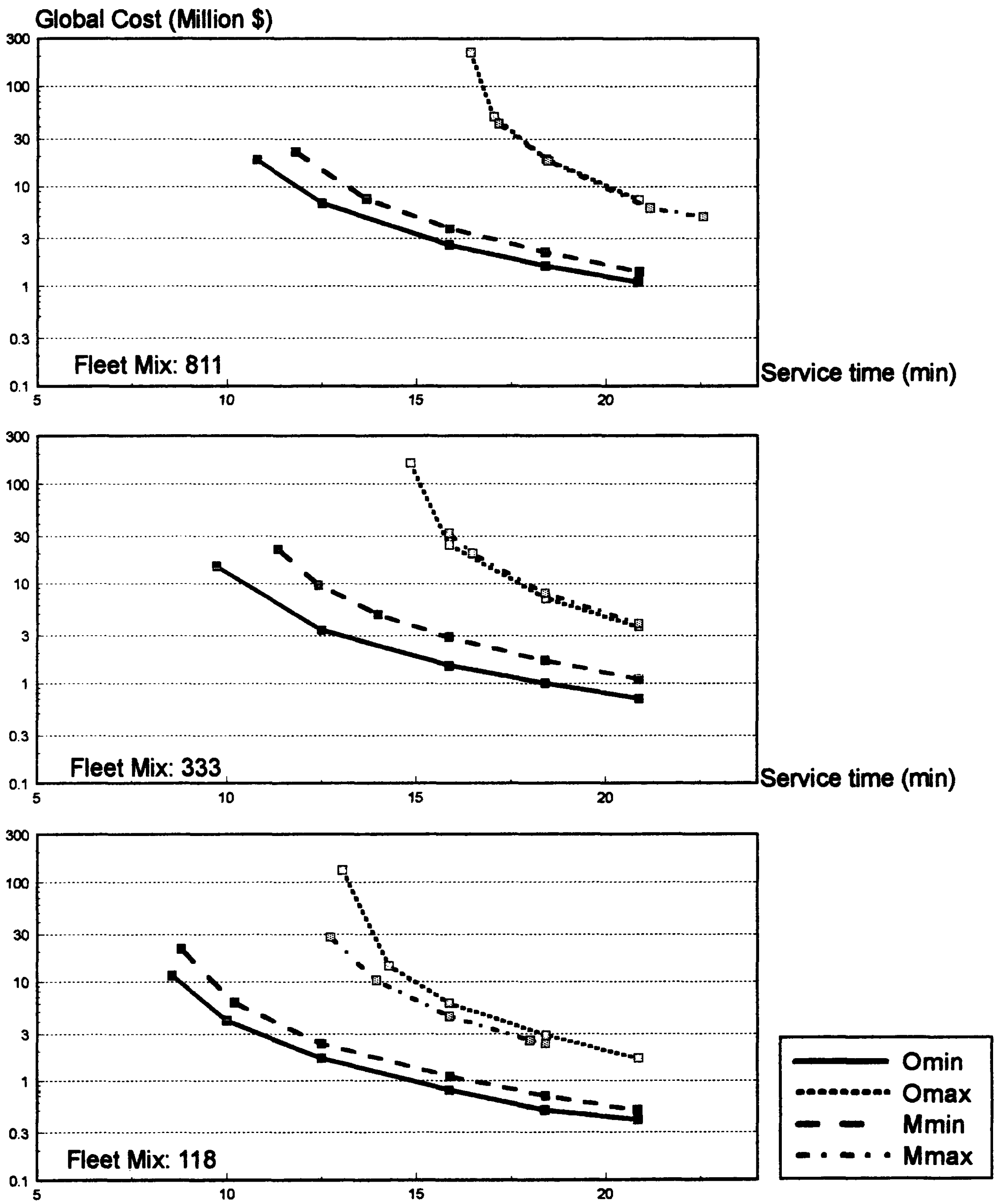
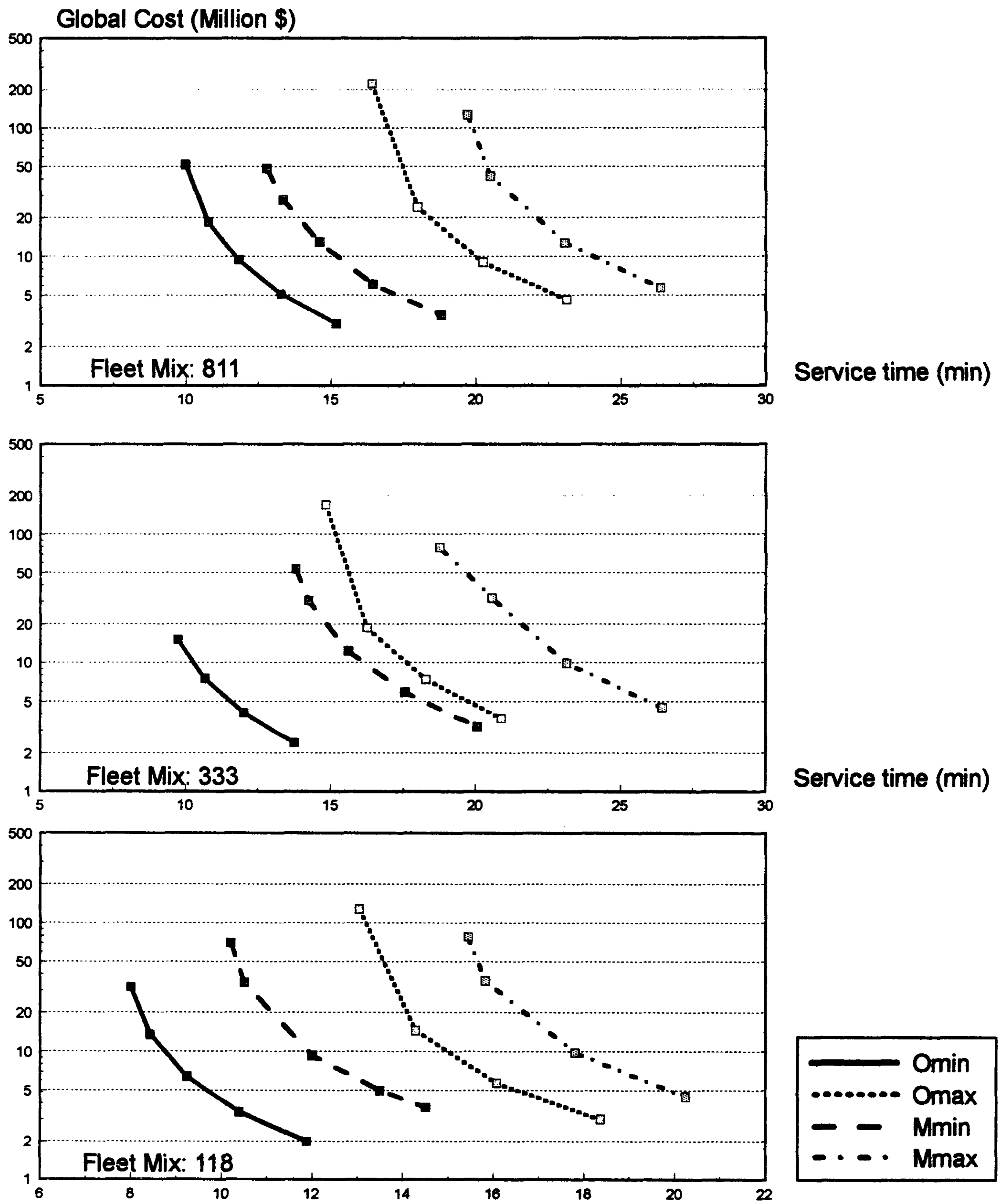


Figure No. 3.9

EFFECT OF SPEED MODE ON GLOBAL COST
FCFS ROUTINE 2-WAY, 10 NM2



channel lengths. It can be easily appreciated that the tendencies are the same for all variables for the various fleet mixes. The best strategy is, as far as speed mode and separation distance are concerned, is the one speed-minimum distance (Omin) mode followed by multi speed-minimum distance mode (Mmin). Again, the results show that the most important factor deciding capacity with this model is the separation distance. The difference in results between minimum and maximum distance for the different velocity modes is notable. For longer channels there are a more marked difference between multi and one-speed modes while the opposite is true for shorter channels, under the said conditions. Note that even when reducing the separation distance improves service times, global costs and global waiting times, the effect is more pronounced as the channel gets longer.

If the maximum separation and multi-speed mode (Mmax) is used as base case for a 5-NM long channel, not only do a Omin strategy result in a much lower service rate as discussed before, but on much lower global waiting time (Figure 3.5) for the same service level. For an arrival rate of a ship every 17.5 min, the Omin mode results in about 6 min of average global waiting, while the Mmin mode results in about 9 min and the Omax and Mmax modes in about 110 min for a 811 fleet mix. This means the average vessel on a Mmax mode has to wait 18 times more than one on a Omin mode and 12 times more than one on a Mmin mode. Other fleet mixes do not result in such dramatic reduction but show marked improvement nonetheless. For a 333 fleet mix and the same service level, waiting on a Mmax mode is about eight times higher than waiting on a Omin mode and about six times higher than waiting on a Mmin mode (Omax waiting time is very similar). For a 118 fleet mix, waiting under a Mmax mode is seven times that of an Omin mode. Global costs show similar tendencies (figures 3.7 to 3.9) and similar reduction ratios. These comparison are punctual. Different ratios occur at different congestion level increasing as Congestion increases and decreasing otherwise.

When a shorter channel is reviewed, the same basic tendencies are observed. For instance, a 811 fleet mix under FCFS routine, there are basically no changes in waiting for a Omin or a Mmin combination while there is a 10 to 15% difference between waiting times on Omax and Mmax choices. There is, however, a large difference between choices with minimum separation and those with maximum separation: about 10 times more waiting when the separation is maximum. This trend is conserved for other fleet mixes. When the fleet is 118, the difference between Mmax y Omax is more marked favoring a Omax mode.

For a 10 NM long channel, a one speed strategy is the most appropriate one. As can be appreciated on figure 3.6, the relative difference between Omax and Mmin modes is much lower than on the other channel lengths reviewed. This is true for all fleet mixes included. For the same service rate, say one vessel every 20 min, global waiting times for Omin, Mmin, Omax, Mmax are 5 min, 14 min, 35 min and 180 min respectively. This translates on an improvement of 36, 12 and five times over the base case (Mmax). Again, the relative improvements change with the level of congestion on the channel.

When strategies other than FCFS are considered, the results are similar. Figures 3.10 to 3.15 show model results for the different queuing routines. In all fleet mixes, channels with low congestion rates benefit little from

Figure No. 3.10

EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING

MULTI SPEED - MAXIMUM DISTANCE

2 WAY, 5NM

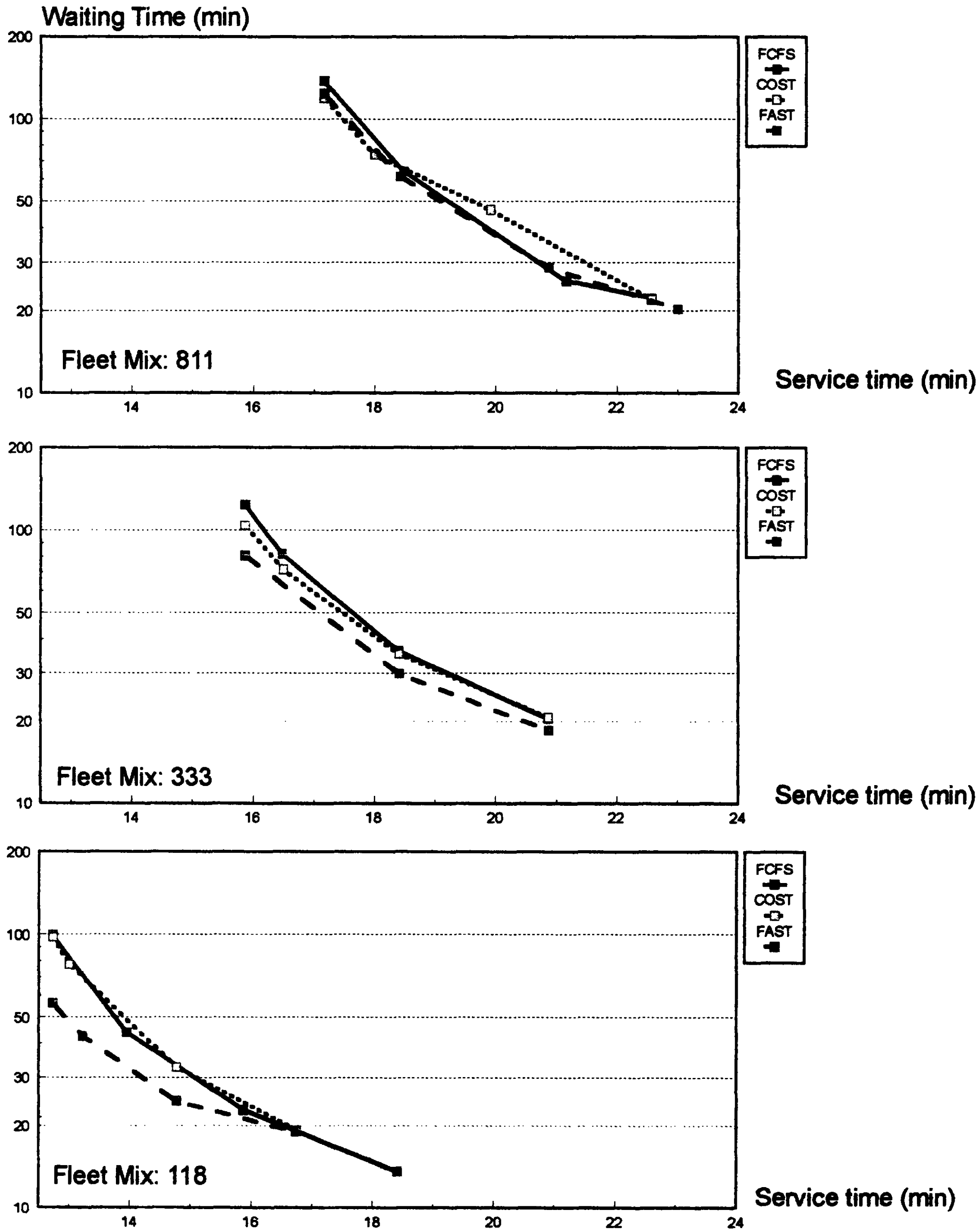


Figure No. 3.11

EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING

MULTI SPEED - MINIMUM DISTANCE

2 WAY, 5 NM

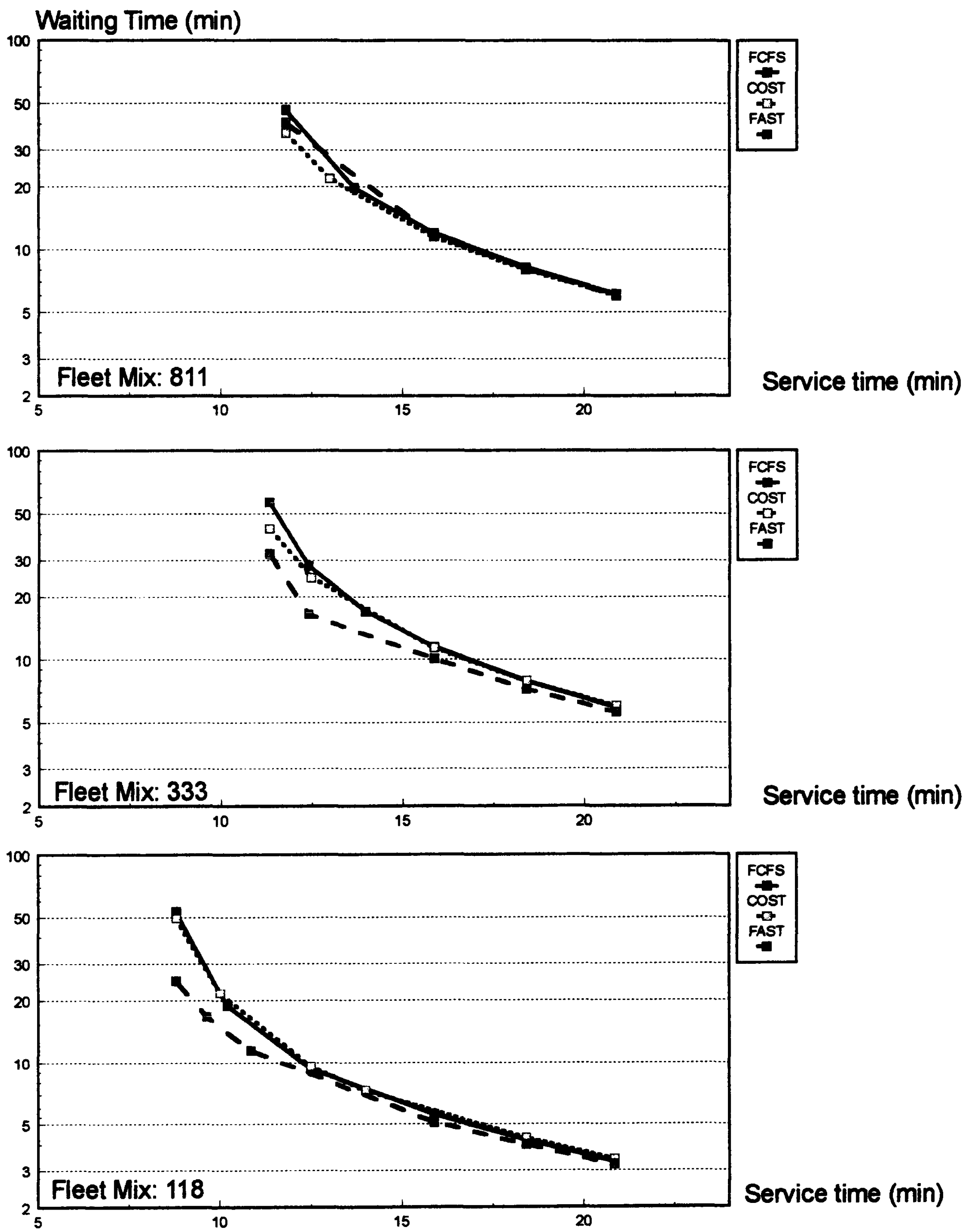


Figure No. 3.13

EFFECT OF QUEUEING ROUTINE IN GLOBAL COST

MULTI SPEED - MAXIMUM DISTANCE

2 WAY, 5 NM

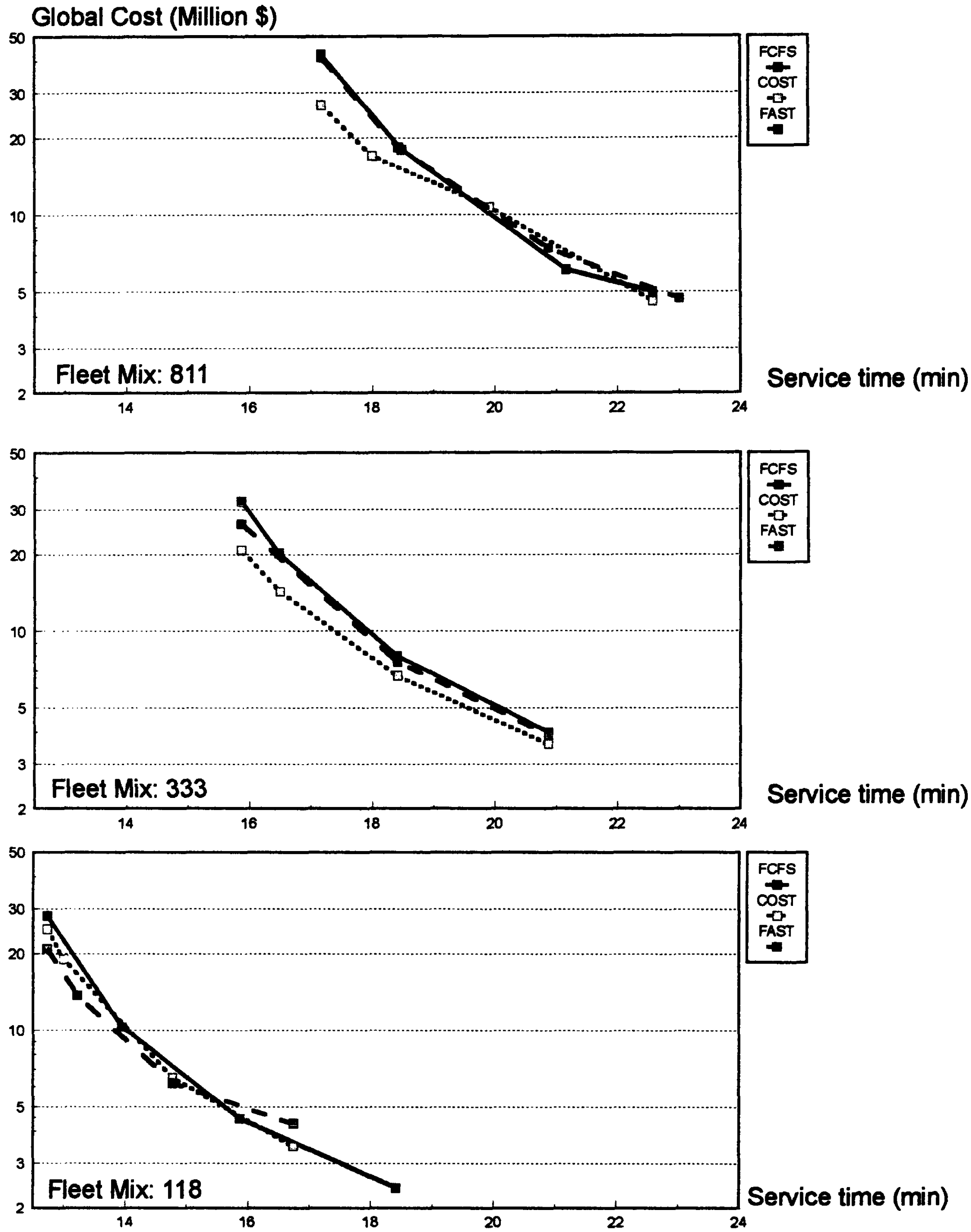


Figure No. 3.14

EFFECT OF QUEUEING ROUTINE IN GLOBAL COST

MULTI SPEED - MINIMUM DISTANCE

2 WAY, 5 NM

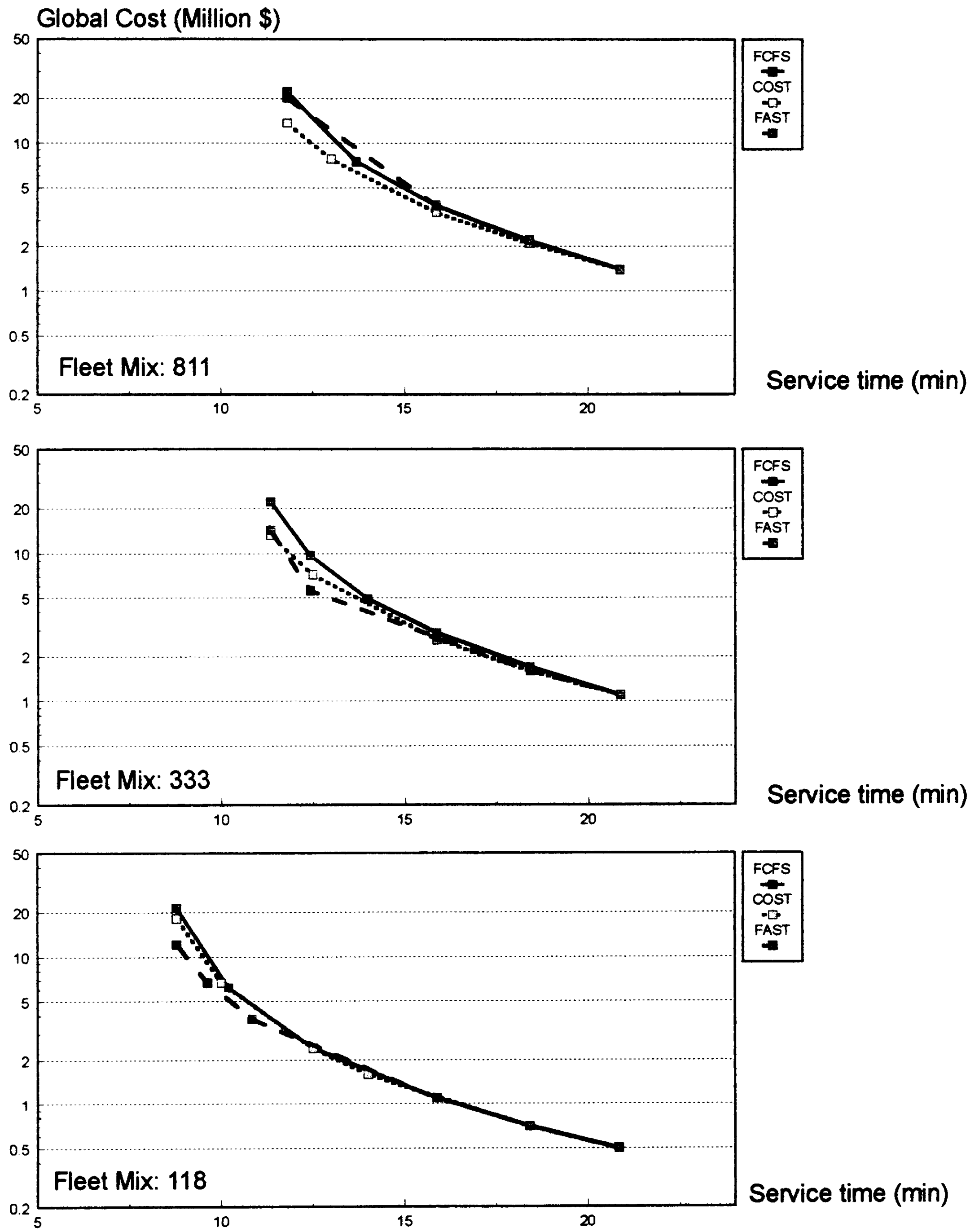
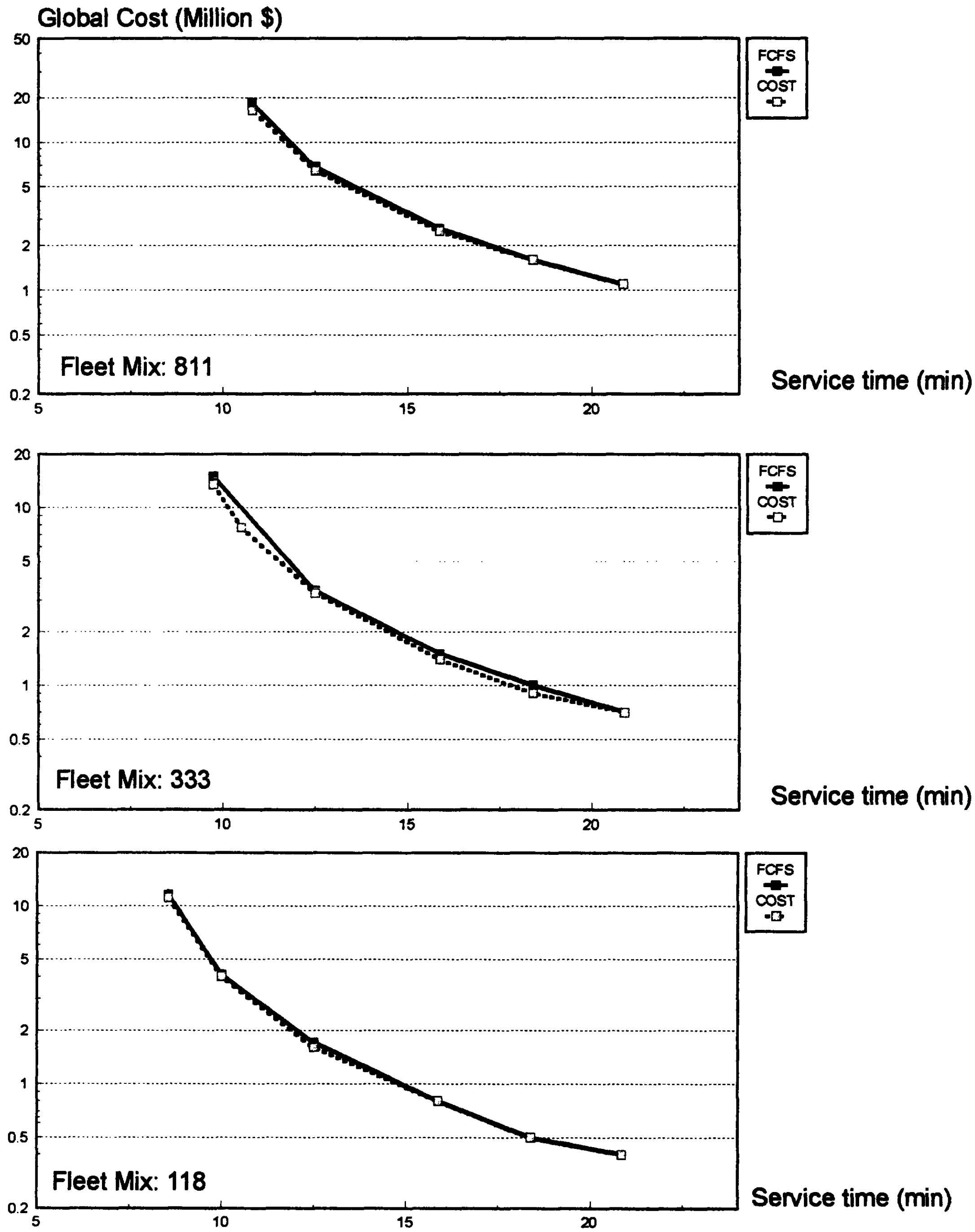


Figure No. 3.15

EFFECT OF QUEUEING ROUTINE IN GLOBAL COST

ONE SPEED - MINIMUM DISTANCE

2 WAY, 5 NM



changes in queuing routine. As congestion rates increase, the benefits of each routine start to show up. These improvements are not as well defined as those found changing separation distance or even speed mode. For instance, FAST strategy succeeds in reducing global waiting time. The level of success is, however, dependent on the fleet mix and the channel length. For a 811 fleet mix, the improvement is almost nil. For a 333 fleet mix, a FAST strategy it results on a 60 min reduction (or about 40%) over a FCFS strategy. For a 118 fleet mix, the reduction is about 40 min (40%) over a FCFS strategy. Comparing global waiting time, a COST routine performs similarly to a FCFS routine (See Appendix C).

When convoys are allowed, the same tendencies apply. The inclusion of convoys as a strategy serves as "fine tuning". For an arrival rate of a ship every 17.5 min under a FCFS strategy, the Omin mode results in about 4 min of average global waiting, while the Mmin mode results in about 7 min and the Omax and Mmax modes in about 100 min for a 811 fleet mix for the optimum convoy length. The difference when an optimum convoy length is not programmed is about 10%. This percentage changes with the convoy length, but waiting times and costs start increasing after a certain convoy length (more than five for this particular fleet mix and channel configuration). The convoys improve on the strong points of COST and FAST strategies, lowering waiting times in a FAST strategy and lowering costs on a COST strategy. Note that improvements with the convoys are not evenly distributed along all pertinent factors. For instance, while a certain convoy length might improve on the waiting time, it does not necessarily improve in the same proportion for the global cost; another convoy length might result on a better global cost but not on the best waiting time even under the same queuing strategy.

3.6 SUMMARY

An easy way to define the best alternative in separation distance and speed mode is through service rate. This index will give an immediate idea of what the channel capacity can be under different circumstances. The use of this index is not limited to a fixed fleet mix so a "dynamic" service rate can be determined as different ships arrive to use the channel.

Separation distance is the most important factor in the definition of capacity. It greatly affects service rates and as shown, service rate defines capacity. Reductions in the separation distance of about 35% results in similar increases in capacity (as defined by the service rate).

The choice of velocity mode depends primarily on the channel length and fleet mix. For short channels, multi-speed mode might be a better than a one speed mode. For longer channels, a one speed mode is likely to be a better one. These observations are based on relative velocity differences between vessels composing the fleet mix. If velocities in the fleet mix have a low variance, a multi-speed strategy will resemble a one speed mode. If the variance is high among vessels' speeds, the difference between a multi-speed mode and a one speed one will be larger and the effect of channel length more acute. The choice of speed mode will, therefore, depend on a channel's particular fleet mix and length.

It has been shown that for low arrival rates, the use of a queuing model does not affect costs and capacity. As congestion increases, the relative importance of a queuing mode also increases. Once the congestion is nearing capacity, the choice of queuing model depends on the objective in mind. If maintaining low cost -while disregarding users priority arrival- is the objective, the best strategy is to use a COST routine. When the prime objective is to allow the highest number of vessels through the channel again disregarding priority of arrival, then the FAST strategy is the optimum one. FCFS strategy does not have any advantages over the other two strategies for high arrival rates. It's only advantage is that it does maintain an arrival priority that might please some channel users. These conclusions are effective when referring to same velocity modes and separations distance.

Convoys improve capacity on any given queuing model. The optimum convoy length varies with the conditions under analysis and the objective in mind. The improvements -although they might be marginal while on low usage levels- become more important as arrivals get near capacity but are not as important as those obtained changing other model factors.

CHAPTER No. 4

4. ECONOMIC CONSIDERATIONS

Now that the potential benefits have been described in the previous chapter, the economic benefits will be defined. The analysis will be limited to two cases: A 5-NM long, 2-way channel and a 10 NM long 2-way channel. Each has been modeled for the speeds and queues described in the previous chapter under an evenly distributed fleet mix. This point will add dollar value to the time savings and the individual vessel costs. At this level, no convoys will be planned.

4.1 CASE: 5-NM LONG, 2-WAY CHANNEL

For the evaluation of the 5 NM long channel, the FCFS queuing strategy with maximum separation and multiple speed allowed will be the basis for comparison. The intership arrival rate will be one vessel every 20 min (or 85% of capacity). Under this scheme, about 52600 vessels will be served during one year. The global waiting cost associated to this level of service is about 12 million $/year. If the speed strategy is changed from multiple speed to one speed, the level of utilization drops to 81% of capacity but the number of vessels served remains constant. In this event, the global waiting cost is 11.9 million $/year. If the same level of capacity is to be maintained, vessels can be received every 19 min (for a total of 55200 vessels/year) with a global waiting cost of about 13.2 million $/year.

Changing the base case now to minimum distance but keeping the same arrival rate, the global waiting cost is 2.5 million $/year. Under this separation distance, the channel would be operating at 55% capacity when vessels arrive every 20 min. To increase the capacity to 85%, a vessel should arrive every 12.8 min (82000 vessels/year). If a vessel is received at this rate, the global waiting cost is 14.6 million $/year.

Finally, changing not only the minimum distance but also the speed mode to one speed, the global waiting cost is 2.2 million $/year. The level of capacity when vessels arrive every 20 min is 53%. The channel can serve a ship every 12.5 min (84000 vessels/year) and operate at 85% capacity. Working at 85% capacity, the global waiting cost would be 9.1 million $/year.

Note that changing solutions always either reduce cost or increase the throughput in the channel in most cases substantially. When arrival levels are maintained, then costs are reduced. Otherwise, when arrivals increase to accommodate the increase capacity, cost increase but the number of vessels served also increases. Cost are reduced by 80% or about 10 million $/year when the distance is minimized and the one speed strategy is in place.

4.2 CASE: 10-NM LONG, 2-WAY CHANNEL

The base case for this channel is the same as the one described on point 4.2. Here, the base service time is 22.7 min (85% capacity for the multi-speed maximum distance case). The channel can serve 46300 vessels in one year and the waiting cost is 21.9 million $/year. When the speed strategy in the channel is changed to one speed, the capacity utilization level drops to 64%. and the waiting cost is 9.1 million \$/year. When the capacity utilization is 85%, this combination results in a waiting cost of 73 million \$/year while serving 61200 vessels.

If the multi speed mode and the arrival rate is maintained while changing the separation distance to the minimum, the global waiting cost is about 3 million \$/year. The utilization level is then 74%. Increasing the arrival rate to 85% of the capacity will result in a global waiting cost of $14.6 * 10^6$ \$/year. The cost is distributed among 73500 vessels that are served.

For the last combination, one-speed and minimum distance with vessel arriving every 22.7 min, the predicted global waiting cost is about 2 million \$/year. This arrival rate corresponds to 45% utilization under these conditions. At 85% capacity, the global cost is 36.5 million \$/year but 102700 vessels can use the facility.

As in the case of a 5-NM long channel, there is always improvement. The improvement is either additional capacity or lower waiting costs. In the most extreme case, the channel manager can either save 20 million \$/year or allow twice as many ships as before the change in strategy is set.

4.3 SUMMARY

Even though the analysis is limited to two cases, the savings in global waiting cost seem to justify investments in a distance reduction system. The savings vary with the strategy selected, channel length and fleet mix. Even though this last factor was not considered in the previous analysis, the service rates described in point 3 indicate such behavior. Again, the financial benefits are the savings in waiting costs or the increased channel usage fees due to the increased traffic.

The dramatic savings described kick in when a channel is nearing capacity. When the level of utilization is lower -say about 70%- the said services might not materialize or become much lower. A case by case analysis must be pursued in order to determine potential savings. Using the model described in the previous chapter this can be determined under a myriad of conditions.

CHAPTER No. 5

5. TRAFFIC MANAGEMENT SYSTEM

All elements discussed in previous chapters make up a navigation system: an access channel, a fleet and a traffic management program. These elements can be taken in conjunction to establish a system for traffic management. In chapter 2, the safety factors in the definition of separation distance and the ship domain were discussed. At that point the feasibility of reducing separation distance was established since information is available to the user. In chapter 3, the system for traffic management and the factors to consider when developing a strategy was established. The proposed model can then be used to define a best approach. The economic benefits of distance reduction were established in chapter 4. In short, it all points to the potential benefits and feasibility of reducing separation distance. This chapter describes a system wide approach to the problem. The objective will be to propose a system than can be used with minimum start up costs and with readily available equipment and know how.

5.1 REQUIREMENTS

Let's check the requirements for the system to work. Under the case to be considered base, a channel is operated with distances as determined by equations 3.2 (maximum distance), a multi-speed mode and a FCFS queuing strategy. To improve on this, the first factor to be considered will be separation distance. As established before, this is feasible if he following condition is met: information is available. From the previous analysis, the only way distance can be reduced is by providing continuous position information to the user.

The proposal is to install a GPS (Global Position System) like set up. Such system installed on a vessel will send a continuous radio signal to a "fixed" land station that will have a dual purpose:

a) improve the accuracy of the "mobil" GPS on the vessel and
b) inform a control center of the vessels position continuously.

The "fixed" land station will monitor all incoming vessels and will send a signal back with the position of all incoming vessels. The system installed on the vessel is connected to a inboard computer that will monitor the vessel position in the channel and will also receive data via modem or radio signal with the position of other vessels in the vicinity.

The accuracy of a GPS system will depend mainly on the number of satellites the device keeps track of. Devices in the market that receive up to 6 satellite signals have accuracy of 1 to 5 m. This level seems more than enough for our ship positioning purposes. Since the signal is sent continuously (a standard systems can send four signals every second), not only is the position known but also the speed of the vessel at any given time. The requirements for the distance reduction are then met.

A computer can be set up so it can pick up signals coming from many vessels. Systems similar to this are already in the market and can monitor as many as 100 vessels. Since the signal priority and effect on the overall system will depend on the distance to the terminal or channel, different signal acceptance rates can be defined according to this factor. In other words, the computer might accept signals every minute from incomming vessels not in the waiting area and every second for vessels using the channel for instance. A fast PC might be able to handle this data with special modifications.

The previously described set up takes care of the information system. The next step is to use the model to come up with the best overall strategy as defined by channel usage policy and costs. This is done with the modeling program described on chapter 3. The program is flexible enough and provides information to make an informed decision on the best overall strategy. Even though, the model does not accept vessel information currently, it can be easily modified to keep track of incoming vessels and provide the channel manager with probable arrival times as it places the vessel in queue (and on its own data base).

5.2 SUMMARY

A system to handle vessel traffic in a navigation channel can be installed with existing off-the-shelve technology and a relatively minor investment. Since all GPS systems signals are not necessarily compatible, a portable computer with modem/FAX (and a cellular phone) connected to a "mobil" GPS might be carried by the pilot to the vessel. This might be a temporary solution for GPS manufacturers are working on universal standards for signal transition and reception.

The "fixed" land system will include a GPS that will serve as reference point to all other positioning systems hooked up to a computer that will monitor the vessels' positions, estimated arrivals and probable waiting times (using the model proposed on this paper). This could be handled by one or several personal computers that can work on a local area network if one is not sufficient to handle all incoming information.

CHAPTER No. 6

6. CONCLUSIONS AND RECOMMENDATIONS

A traffic management system is feasible and easily implemented using existing technology. All constituent parts can be put together with a small compared to the possible savings. The savings in time and costs are relative to the level of usage of the navigation channel. Nevertheless, even when in low usage levels some savings can be made. Once an optimum solution set is chosen for a waterway, the usage level at which a management system can be set up with some profit can be easily calculated.

In general, the model proposed can, by itself, generate savings in time and money. The amount of the savings will depend on the basic objectives of the port manager: cost reduction or waiting time reduction. The capacity of the system can also be increased by the use of the model. The use of convoys can be selectively chosen depending on the mix of vessels in waiting queue, and will also enhance the system objectives. Also, once a model for operation has been established, day to day operations can be managed with the model.

The model results show the financial and operative advantage of reducing separation distance. The definition of this factor needs to be reviewed under the light of the existing positioning, communication and computing technologies.

BIBLIOGRAPHY

1. Abdelgalil, E.M., "Shipping Casualties and Ship's Domain". International Symphosium on Marine Traffic. Liverpool 1978 (p. 95).

2. Agerschou, Hans et al. "Planning and Design of Ports and Marine Terminals". Wiley Interscience. 1983.

3. Andersen, Terje. "Use of Risk Analysis in Port Siting, Planning and Design". Veritas Offshore Technology and Services. Oslo, Norway.

4. Bottoms, Eric. "Practical Tonnage Capacity of Canalyzed Waterways". Journal of the Waterways, Harbors Division. Vol 92. (WW1) Feb. 1966.

5. Bruun, Per. "Port Engineering". 4th ed. Gulf Publishing Company. July 1989.

6. DELFT. "Aspects of Navigability of Constrained Waterways, including Harbour Entrances. III Symposium Delft, 1978.

7. Frankel, Ernst G., "Port Design and Development". John Wiley Interscience, New York, 1987.

8. Eden, Edwin. "Vessels Controllability in Restricted Waters". Journal of Waterways, Harbors and Coastal Engineering. Vol 97 (1971). WW3.

9. Hergert, Douglas. "Visual Basic 3.0 Programming with Windows Applications". 2nd ed. Random House, 1993.

10. International Symphosium on Marine Traffic. " Selection of a Marine Traffic Control System". Proceedings. 1977 (p. 170)

11. International Symphosium on Maritime Communications and Control. "The Use of a Mathematical Model in a Collision Avoidance System". Proceedings. London November 1990.

12. Keinrock, Leonard. "Queing Systems, Volume I: Theory". John Wiley & Sons, Inc. New York 1975.

13. Kemp, John and Goodwin, Elizabeth. "Risk Analysis Within the COST 301 Project". City of London Polytechnic.

14. Kray, Casimir. "Design of Ship Channels and Maneuvering Areas". Journal of the Waterways, Harbors and Coastal Engineering. Feb. 1973 (WW1).

15. Mansfield, Richard. "The Visual Guide to Visual Basic for Windows". 2nd ed. Ventana Press, 1993.

16. O'Reilly, Jean and Lilegdon, William. "SLAM II Quick Reference Manual". Pritsker & Associates, Inc. West Lafayette, Indiana, 1987.

17. PIANC. "Optimal Layout Dimensions of Maritime Fairways in Shallow Seas, Sea Straits and Maritime Waterways". Recommendations by PIANC for the Reception of Large Vessels Group IV. Bulletin # 35.

18. PIANC. "Capability of Ship Manoeuvring Simulation Models for Approach Channels and Fairways in Harbours". Report of Working Group No. 20 of

Permanent Technical Committee II. Supplement to Bulleting No. 77 (1992).

19. Pristsker, Alan. "Introduction to Simulation and SLAM II, 3ed. John Wiley & Sons, Inc, New York, 1986.

20. Soo Lim, Chin. "Channel Capacity Models: Strategic Use of Navigational Channels". Doctor of Philosophy Thesis. Massachusetts Institute of Technology. 1989.

21. Thorensen, Carl A. "PORT DESIGN. Guidelines and Recommendation"s. Tapir Publishers. 1988.

APPENDIX A

PROGRAMMING CODE

BASICDAT.FRM

```
Dim NewChannelLength As Integer
Dim ControlShip As Integer

Sub DrawFrameOn (TopLeftControl As Control, LowestRightControl As Control, Style As S
tring, FrameWidth)
   DW = DrawWidth
   FS = FillStyle
   SM = ScaleMode
   DrawWidth = 1
   FillStyle = 1
   ScaleMode = 1
   ST$ = LCase$(Left$(Style$, 1))
   Lft = TopLeftControl.Left
   TopLft = TopLeftControl.Top
   Hite = TopLeftControl.Height
   Rite = LowestRightControl.Left + LowestRightControl.Width
   RiteBotm = LowestRightControl.Top + LowestRightControl.Height
   ' Use tallest control as Y
   If RiteBotm > Hite Then Hite = RiteBotm
   ' Draw a Thick Box
   Line (Lft - FrameWidth, TopLft - FrameWidth)-(Rite + FrameWidth, RiteBotm + FrameWidth),
QBColor(7), BF
   ' Draw Highlight and Shadow lines
   it = 15
   RB = 8
   If ST$ = "i" Then it = 8: RB = 15
   Line (Lft - FrameWidth, TopLft - FrameWidth)-(Rite + FrameWidth, TopLft - FrameWidth),
QBColor(it)
   Line (Lft - FrameWidth, TopLft - FrameWidth)-(Lft - FrameWidth, Hite + FrameWidth),
QBColor(it)
   Line (Rite + FrameWidth, TopLft - FrameWidth)-(Rite + FrameWidth, RiteBotm +
FrameWidth), QBColor(RB)
   Line (Rite + FrameWidth, RiteBotm + FrameWidth)-(Lft - FrameWidth, Hite + FrameWidth),
QBColor(RB)
   If ST$ <> "i" Then
      Line (Lft - FrameWidth - 25, TopLft - FrameWidth - 25)-(Rite + FrameWidth + 10
, RiteBotm + FrameWidth + 10), QBColor(0), B
   End If
   DrawWidth = DW
   FillStyle = FS
```

```
   ScaleMode = SM
End Sub

Sub Emboss (DoWhat As Control, Style As String, FrameSize As Integer)
   DrawWidth = 1
   ' Embossed=1; Etched=0 (inner or outer respectively)
   If Left$(LCase$(Style), 1) = "o" Then
     cg = 15
     c2 = 8
   Else
     cg = 8
     c2 = 15
   End If
   ' Set the frame distances relative to control
   DoWhat.BackColor = QBColor(7)
   X = DoWhat.Left - 46 - FrameSize
   Y = DoWhat.Top - 46 - FrameSize
   X1 = DoWhat.Left + DoWhat.Width + 26 + FrameSize
   Y1 = DoWhat.Top + DoWhat.Height + 26 + FrameSize
   ' Draw first frame
   Line (X, Y)-(X1, Y1), QBColor(cg), B
   ' Draw second frame down and to the right
   Line (X + 14, Y + 14)-(X1 + 20, Y1 + 20), QBColor(c2), B
   ' Add dost to make two of the corners look smooth
   ' only for embossed style
   If cg = 15 Then
     PSet (X1, Y + 14), QBColor(cg)
     PSet (X + 14, Y1), QBColor(cg)
   End If
End Sub

Sub NextS_Click ()
   If ControlShip + 1 = 4 Then
     ControlShip = 1
   Else
     ControlShip = ControlShip + 1
   End If
   Text1.Text = "  " + Format$(ControlShip, "##")
   Text2.Text = "   " + Format$(SType(ControlShip).DWT, "######")
   Text3.Text = "  " + Format$(SType(ControlShip).Speed, "###.##")
   Text4.Text = "  " + Format$(SType(ControlShip).Length, "###.##")
   Text5.Text = "  " + Format$(SType(ControlShip).WCost * 60, "####.##")
End Sub

Sub PreviousS_Click ()
   If ControlShip - 1 = 0 Then
     ControlShip = 3
   Else
     ControlShip = ControlShip - 1
```

```
    End If
    Text1.Text = "  " + Format$(ControlShip, "##")
    Text2.Text = "   " + Format$(SType(ControlShip).DWT, "######")
    Text3.Text = "  " + Format$(SType(ControlShip).Speed, "###.##")
    Text4.Text = "  " + Format$(SType(ControlShip).Length, "###.##")
    Text5.Text = "  " + Format$(SType(ControlShip).WCost * 60, "####.##")
End Sub

Sub ShowBasicData ()
    'OkCD.Caption = "Edit"
    Printer.Enabled = True
    ControlShip = 1
    ' Sets variables in screen
    ' according to user's data
    ' Sets channel data
    ChLength.Text = " " + Format(ChannelLength, "######")
    CycleT = "  " + Format(CycleTime, "######")
    If OneLane Then
      OneWay.Value = True
      TwoWay.Value = False
    Else
      OneWay = False
      TwoWay.Value = True
    End If
    If OneLane Then
      Label10.Visible = True
      CycleT.Visible = True
    Else
      Label10.Visible = False
      CycleT.Visible = False
    End If
    ' Sets ship data
    Text1.Text = "  " + Format$(ControlShip, "##")
    Text2.Text = "   " + Format$(SType(ControlShip).DWT, "######")
    Text3.Text = "  " + Format$(SType(ControlShip).Speed, "###.##")
    Text4.Text = "  " + Format$(SType(ControlShip).Length, "###.##")
    Text5.Text = "  " + Format$(SType(ControlShip).WCost * 60, "####.##")
End Sub


Sub CancelCD_Click ()
    Unload frmBasicData
End Sub

Sub ChLength_Change ()
    If OkCD.Caption = "Save" Then
      ChannelLength = Val(ChLength)
    End If
End Sub
```

```
Sub cmdOk_Click ()
   Unload frmBasicData
End Sub

Sub CycleT_Change ()
   ' Changes value of cycle time
   If OkCD.Caption = "Save" Then
     CycleTime = Val(CycleT)
   End If
End Sub

Sub Form_Load ()
   frmBasicData.Top = 100
   frmBasicData.Left = 2250
   frmBasicData.Height = 4725
   frmBasicData.Width = 5775

   ' Opens file with basic data
   OpenBasicData
   ' Sets basic controls and buttoms
   ShowBasicData
End Sub

Sub OKCD_Click ()
   ' Accepts change of channel
   ' characteristics if requested
   If OneWay Then
      Label10.Visible = True
      CycleT.Visible = True
   End If
     If OneWay Then
       CycleT.Visible = True
       Label10.Visible = True
     Else
       CycleT.Visible = False
       Label10.Visible = False
     End If
     ChannelLength = Val(ChLength.Text)
     OneLane = OneWay.Value
     ' Saves new data
     SaveBasicData
End Sub

Sub OneWay_Click ()
   Label10.Visible = True
   CycleT.Visible = True
End Sub

Sub spiShips_SpinDown ()
```

```
    If ControlShip + 1 = 4 Then
      ControlShip = 1
    Else
      ControlShip = ControlShip + 1
    End If
    Text1.Text = "  " + Format$(ControlShip, "##")
    Text2.Text = "   " + Format$(SType(ControlShip).DWT, "######")
    Text3.Text = "  " + Format$(SType(ControlShip).Speed, "###.##")
    Text4.Text = "  " + Format$(SType(ControlShip).Length, "###.##")
    Text5.Text = "  " + Format$(SType(ControlShip).WCost * 60, "####.##")
End Sub

Sub spiShips_SpinUp ()
    If ControlShip - 1 = 0 Then
      ControlShip = 3
    Else
      ControlShip = ControlShip - 1
    End If
    Text1.Text = "  " + Format$(ControlShip, "##")
    Text2.Text = "   " + Format$(SType(ControlShip).DWT, "######")
    Text3.Text = "  " + Format$(SType(ControlShip).Speed, "###.##")
    Text4.Text = "  " + Format$(SType(ControlShip).Length, "###.##")
    Text5.Text = "  " + Format$(SType(ControlShip).WCost * 60, "####.##")
End Sub

Sub Text2_Change ()
    ' Changes value of DWT
    If OkCD.Caption = "Save" Then
      SType(Val(Text1.Text)).DWT = Val(Text2.Text)
    End If
End Sub

Sub Text3_Change ()
    ' Changes value of Speed
    If OkCD.Caption = "Save" Then
      SType(Val(Text1.Text)).Speed = Val(Text3.Text)
    End If
End Sub

Sub Text4_Change ()
    ' Changes value of Length
    If OkCD.Caption = "Save" Then
      SType(Val(Text1.Text)).Length = Val(Text4.Text)
    End If
End Sub

Sub Text5_Change ()
    ' Changes value of Speed
    If OkCD.Caption = "Save" Then
```

```
        SType(Val(Text1.Text)).WCost = Val(Text5.Text) / 60
    End If
End Sub

Sub TwoWay_Click ()
    Label10.Visible = False
    CycleT.Visible = False
End Sub
```

EXPER.FRM

```
Sub Panel3D1_Click ()
   Panel3D2.FloodPercent = Panel3D2.FloodPercent + 2
End Sub

Sub cmdCancel_Click ()
    Unload frmFile
End Sub

Sub cmdOk_Click ()
   If FileAction = True Then
      FiletoSave = lblDirectory.Caption & "\" & txtFile.Text
   Else
      FiletoOpen = lblDirectory.Caption & "\" & txtFile.Text
   End If
   Unload frmFile
End Sub

Sub dirDirectory_Change ()
   filFile.Path = dirDirectory.Path
   lblDirectory.Caption = filFile.Path
   filFile.ListIndex = 0
End Sub

Sub drvDrive_Change ()
   dirDirectory.Path = drvDrive.Drive
End Sub

Sub filFile_Click ()
   txtFile.Text = filFile.List(filFile.ListIndex)
End Sub

Sub Form_Load ()
   dirDirectory.Path = drvDrive.Drive
   filFile.Path = dirDirectory.Path
   lblDirectory.Caption = filFile.Path
   filFile.ListIndex = 0
   txtFile.Text = filFile.List(filFile.ListIndex)

End Sub
```

GENSHPDA.FRM

```
Sub ShowBasicData ()
   ' Togles variables off
   ShipArrival.Enabled = False
   Ship1Perc.Enabled = False
   Ship2Perc.Enabled = False
   Ship3Perc.Enabled = False
   Text1.Enabled = False
   ShipArrival.Text = ArrivalRate
   Ship1Perc.Text = Format(Ship1, "##.##")
   Ship2Perc.Text = Format(Ship2, "##.##")
   Ship3Perc.Text = Format(Ship3, "##.##")
   Text1.Text = SimDays
End Sub

Sub CancelPD_Click ()
   If EditPD.Enabled = False Then
     ' Reactives all keys
     frmGenShpData.Enabled = True
     EditPD.Enabled = True
     HelpP.Enabled = True
     frmGenShpData.Enabled = True
     ShipArrival.Visible = True
     Ship1Perc.Visible = True
     Ship2Perc.Visible = True
     Ship3Perc.Visible = True
     Text1.Visible = True
     Label2.Visible = True
     Label3.Visible = True
     Label4.Visible = True
     Label5.Visible = True
     Label6.Visible = True
     Label7.Visible = True
     Label8.Visible = True
     Label9.Visible = True
     Label10.Visible = True
     chkSaveFile.Visible = True
   Else
     If EditPD.Caption = "Edit" Then
       ' On edit mode, takes form
       ' off screen
       Unload frmGenShpData
     Else
       ' On save mode, changes
       ' data back to initial
       EditPD.Caption = "Edit"
       GenArrival.Enabled = True
```

```
      ShowBasicData
    End If
  End If
End Sub

Sub EditPD_Click ()
  If EditPD.Caption = "Edit" Then
    EditPD.Caption = "Save"
    ' Togles variables off
    GenArrival.Enabled = False
    ShipArrival.Enabled = True
    Ship1Perc.Enabled = True
    Ship2Perc.Enabled = True
    Ship3Perc.Enabled = True
    Text1.Enabled = True
  Else
    EditPD.Caption = "Edit"
    ' Togles variables off
    GenArrival.Visible = True
    GenArrival.Enabled = True
    ShipArrival.Enabled = False
    Ship1Perc.Enabled = False
    Ship2Perc.Enabled = False
    Ship3Perc.Enabled = False
    Text1.Enabled = False
    ' Saves data changed
    SaveBasicData
  End If
End Sub

Sub Form_Load ()
  frmGenShpData.Top = 100
  frmGenShpData.Left = 2250
  frmGenShpData.Height = 4695
  frmGenShpData.Width = 4275

  ' Loads basic data
  OpenBasicData
  ' Shows basic data
  ShowBasicData
End Sub

Sub GenArrival_Click ()
  ' Allows only cancel key
  GenArrival.Enabled = False
  EditPD.Enabled = False
  HelpP.Enabled = False
  frmGenShpData.Enabled = False
  ' Creates data base
```

```
    GenerateArrival
    ' Reactives all keys
    frmGenShpData.Enabled = True
    EditPD.Enabled = True
    HelpP.Enabled = True
    frmGenShpData.Enabled = True
End Sub

Sub Ship1Perc_Change ()
    ' Sets value of ship1
    Ship1 = Val(Ship1Perc.Text)
    ' Changes value of third vessels
    ' depending on first and second
    Ship3Perc.Text = 100 - (Ship1 + Ship2)
End Sub

Sub Ship2Perc_Change ()
    ' Sets value of ship2
    Ship2 = Val(Ship2Perc.Text)
    ' Changes value of third vessels
    ' depending on first and second
    Ship3Perc.Text = 100 - (Ship1 + Ship2)
End Sub

Sub Ship3Perc_Change ()
    ' Sets value of ship3
    Ship3 = Val(Ship3Perc.Text)
    ' Changes value of third vessels
    ' depending on first and second
    Ship1Perc.Text = 100 - (Ship3 + Ship2)
End Sub

Sub ShipArrival_Change ()
    ' Sets arrival rate
    ArrivalRate = Val(ShipArrival.Text)
End Sub

Sub Text1_Change ()
    ' Sets length of simulation
    SimDays = Val(Text1.Text)
End Sub
```

MDIPRINC.FRM

```
Sub cmdSalida_Click ()
   End
End Sub

Sub BasicData_Click ()
   BasicDataRoutine
End Sub

Sub ExitProg_Click ()
   ' Ends program
   msg = "This will end your Channel Operations Optimization session."
   Answer = MsgBox(msg, 65, "Quit")
   If Answer = 1 Then
      End
   End If
End Sub

Sub GenDB_Click ()
   ' Calls routine to generate data
   GenShpArrival
End Sub

Sub MDIForm_Load ()
   OpenBasicData
   OpenShipDb
   Load frmOperations
   frmOperations.Show

   On Error GoTo ErrorTrap
   ' Initializes necesary files
   ' Checks if User ship Data Base exits
   XFile = Dir$("C:\simmod\Ship235.Txt")
   If UCase(XFile) <> UCase("Ship234.Txt") Then
      ' File does no exist
      ' Generates new
      FileCopy "c:\simmod\blank.txt", "c:\simmod\Ship234.Txt"
   End If
GoTo Jump
ErrorTrap:
   ErrorTrap
Jump:
End Sub

Sub mnuBasicData_Click ()
   BasicDataRoutine
```

```
End Sub

Sub RunSimulation_Click ()
    ' Run simulation
    Process
End Sub

Sub SetupProcess_Click ()
    ' Runs routine for setting up simulation
    SetUpSimulation
End Sub

Sub UserDB_Click ()
    ' Calls user data routine
    UseShpArrival
End Sub
```

MONITOR.FRM

```
Sub Command3D1_Click ()
End
End Sub

Sub Command3D2_Click ()
   gauMonitor.Value = gauMonitor.Value + 2
End Sub

Sub gauMonitor_Change ()
   If gauMonitor.Value >= 100 Then
      Unload frmMonitor
   End If
End Sub
```

OPERATIO.FRM

```
Sub cmdBasicData_Click ()
   Load frmShowBasicData
   frmShowBasicData.Show
   'BasicDataRoutine
End Sub

Sub cmdExit_Click ()
   msg = "This will end your Channel Operations Optimization session."
   Answer = MsgBox(msg, 65, "Quit")
   If Answer = 1 Then
      End
   End If
End Sub

Sub cmdRun_Click ()
   ' Run simulation
   'frmMonitor.Show
   Load frmRunProcess
   frmRunProcess.Show
   'Process
End Sub

Sub cmdSetup_Click ()
   ' Runs routine for setting up simulation
   SetUpSimulation
End Sub

Sub Form_Load ()
   frmOperations.Top = 100
   frmOperations.Left = 100
End Sub
```

OUTPUT.FRM

```
Sub Command3D1_Click ()
   aniMonitor.Frame = 3
   aniPrinter.Frame = 5
   aniFile.Frame = 4
End Sub

Sub Command3D2_Click ()
   aniMonitor.SpecialOp = 1
End Sub

Sub aniMonitor_Click ()
   ' cmdMonitor_Click
End Sub

Sub cmdCancel_Click ()
   Unload frmOutput
End Sub

Sub cmdFile_Click ()
   If SendtoFile = False Then
      aniFile.SpecialOp = 1
      SendtoFile = True
      Load frmFile
      frmFile.Caption = "File for Output"
      FileAction = True
      frmFile.Show
      frmFile.SetFocus
      cmdFile.Caption = "- On -"
   Else
      aniFile.SpecialOp = 1
      SendtoFile = False
      cmdFile.Caption = "- Off -"
   End If
End Sub

Sub cmdMonitor_Click ()
   If SendtoScreen = False Then
      aniMonitor.SpecialOp = 1
      SendtoScreen = True
      cmdMonitor.Caption = "- On -"
   Else
      aniMonitor.SpecialOp = 1
      SendtoScreen = False
      cmdMonitor.Caption = "- Off -"
   End If
```

```
End Sub

Sub cmdOk_Click ()
   If SendtoPrinter = False And SendtoFile = False And SendtoScreen = False Then
      msg = "At least one output device must be selected."
      MsgBox msg, 16, "Output Device"
   Else
      Unload frmOutput
   End If
End Sub

Sub cmdPrinter_Click ()
   If SendtoPrinter = True Then
      aniPrinter.SpecialOp = 1
      SendtoPrinter = False
      cmdPrinter.Caption = "- Off -"
   Else
      aniPrinter.SpecialOp = 1
      SendtoPrinter = True
      cmdPrinter.Caption = "- On -"
   End If
End Sub

Sub Form_Load ()
   If SendtoScreen = True Then
      aniMonitor.SpecialOp = 1
   Else
      aniMonitor.Frame = 1
   End If
   If SendtoPrinter = True Then
      aniPrinter.SpecialOp = 1
   Else
      aniPrinter.Frame = 1
   End If
   If SendtoFile = True Then
      aniFile.SpecialOp = 1
   Else
      aniFile.Frame = 1
   End If
End Sub
```

PRINCIPM.FRM

```
Sub BasicData_Click ()
   BasicDataRoutine
End Sub

Sub ExitProg_Click ()
   ' Ends program
   End
End Sub

Sub Form_Load ()
   ' Initializes necesary files
   ' Checks if User ship Data Base exits
   XFile = Dir$("C:\simmod\Ship235.Txt")
   If UCase(XFile) <> UCase("Ship234.Txt") Then
      ' File does no exist
      ' Generates new
      FileCopy "c:\simmod\blank.txt", "c:\simmod\Ship234.Txt"
   End If
End Sub

Sub GenDB_Click ()
   ' Calls routine to generate data
   GenShpArrival
End Sub

Sub RunSimulation_Click ()
   ' Run simulation
   Process
End Sub

Sub SetupProcess_Click ()
   ' Runs routine for setting up simulation
   SetUpSimulation
End Sub

Sub UserDB_Click ()
   ' Calls user data routine
   UseShpArrival
End Sub
```

PROCESSD.FRM

```
Dim Control As Integer


Sub DisableCell ()
   ' Does not allow user to
   ' change data on screen
   Option3D1.Enabled = False
   Option3D2.Enabled = False
   check3d1.Enabled = False
   Check3D2.Enabled = False
   ChecK3D3.Enabled = False
   Queue1.Enabled = False
   Queue2.Enabled = False
   Queue3.Enabled = False
   Speed1.Enabled = False
   Speed0.Enabled = False
   Separ1.Enabled = False
   Separ2.Enabled = False
   StartDate.Enabled = False
   NDays.Enabled = False
   If Simulation = True Then
     StartDate.Visible = False
     Label8.Visible = False
   End If
   PIcon1.Enabled = True
   PIcon2.Enabled = True
   PIcon3.Enabled = True
   PIcon4.Enabled = True
   Combo1.Enabled = False
End Sub

Sub IniData ()
   ' Changes data to screen format
   ' Fills Combo box
   Combo1.AddItem "No"
   Combo1.AddItem "2"
   Combo1.AddItem "3"
   Combo1.AddItem "4"
   Combo1.AddItem "5"
   Combo1.AddItem "6"
   Combo1.AddItem "7"
   Combo1.AddItem "8"
   Combo1.AddItem "9"
   ' Queue
   If QueueFCFS Then
     Queue1.Value = True
   Else
```

```
      Queue1.Value = False
   End If
   If QueueCOST Then
      Queue2.Value = True
   Else
      Queue2.Value = False
   End If
   If QueueFAST Then
      Queue3.Value = True
   Else
      Queue3.Value = False
   End If
   ' Speed
   If SpeedOne Then
      Speed0.Value = True
   Else
      Speed0.Value = False
   End If
   If SpeedMulti Then
      Speed1.Value = True
   Else
      Speed1.Value = False
   End If
   ' Separation
   If MinSepar = True Then
      Separ1.Value = True
   Else
      Separ2.Value = True
   End If
   ' Date and horizon
   StartDate.Text = IniDate
   NDays.Text = SimDays
   ' Convoy
   If Convoy = 0 Then
      Combo1.Text = "No"
   Else
      Combo1.Text = Convoy
   End If
End Sub

Sub cmdCancel_Click ()
      Unload frmProcessData
End Sub

Sub cmdClearAll_Click ()
   Queue1.Value = False
   QueueFCFS = False
   Queue2.Value = False
   QueueCOST = False
```

```
   Queue3.Value = False
   QueueFAST = False
   Speed0.Value = False
   SpeedOne = False
   Speed1.Value = False
   SpeedMulti = False
   Separ1.Value = False
   Separ2.Value = False
   MinSepar = False
   SendtoFile = False
   SendtoPrinter = False
   SendtoMonitor = False
End Sub

Sub cmdOk_Click ()
   Unload frmProcessData
End Sub

Sub cmdOutput_Click ()
   Load frmOutput
   frmOutput.Show
End Sub

Sub cmdSaveSetup_Click ()
      Control = 0
      Convoy = Val(Combo1.Text)
      If Convoy = 0 Then Convoy = 1
      IniDate = StartDate.Text
      SimDays = Val(NDays.Text)
      ' Opens file and saves data on screen
      SaveBasicData
End Sub

Sub cmdSCriteria_Click ()
   Load frmSelectCriteria
   frmSelectCriteria.Show
End Sub

Sub Form_Load ()
   frmProcessData.Height = 6180
   frmProcessData.Top = 100
   frmProcessData.Left = 2250
   frmProcessData.Width = 5640
   Control = 0
   ' Opens basic data file
   OpenBasicData
   ' Defines values that will show on screen
   IniData
   ' Defines status of cells
```

```
    ' Shows Message
    Message1.Visible = True
    Message1.Caption = "Best combination included when several options indicated"
    ' Assigns values to form varibles
    If SendToScreen Then
      check3d1.Value = 1
    Else
      check3d1.Value = 0
    End If
    If SendtoPrinter Then
      Check3D2.Value = 1
    Else
      Check3D2.Value = 0
    End If
    If SendtoFile Then
      ChecK3D3.Value = 1
    Else
      ChecK3D3.Value = 0
    End If
    If Simulation = True Then
      Option3D1.Value = False
      Option3D2.Value = True
      ' If Simulation, Initial date relevant
      Label8.Visible = True
      StartDate.Visible = True
    Else
      Option3D1.Value = True
      Option3D2.Value = False
      ' If Modeling, Initial date irrelevant
      Label8.Visible = False
      StartDate.Visible = False
    End If
End Sub

Sub NDays_Change ()
    ' Assigns value in box to
    ' variable Time Horizon
    TimeHorizon = Val(NDays.Text)
End Sub

Sub Option3D1_Click (Value As Integer)
    ' Changes Icon and updates
    If Option3D1.Value = False Then
      PIcon1.Visible = True
      PIcon2.Visible = False
      PIcon3.Visible = False
      PIcon4.Visible = True
      ' If Simulation, Initial date relevant
      Label8.Visible = True
```

```
      StartDate.Visible = True
    Else
      PIcon1.Visible = True
      PIcon2.Visible = True
      PIcon3.Visible = True
      PIcon4.Visible = False
      ' If Modeling, Initial date irrelevant
      Label8.Visible = False
      StartDate.Visible = False
    End If
    Simulation = Option3D2.Value
End Sub

Sub Option3D2_Click (Value As Integer)
    ' Changes Icon and updates
    If Option3D1.Value = False Then
      PIcon1.Visible = True
      PIcon2.Visible = False
      PIcon3.Visible = False
      PIcon4.Visible = True
      ' If Simulation, Initial date relevant
      Label8.Visible = True
      StartDate.Visible = True
    Else
      PIcon1.Visible = False
      PIcon2.Visible = True
      PIcon3.Visible = True
      PIcon4.Visible = False
      ' If Modeling, Initial date irrelevant
      Label8.Visible = False
      StartDate.Visible = False
    End If
    Simulation = Option3D2.Value
End Sub

Sub Queue1_Click (Value As Integer)
    ' Checks if another queue mode has been specified
    ' if so allows change
    If Queue1.Value = True Then
      QueueFCFS = True
    Else
      QueueFCFS = False
    End If
End Sub

Sub Queue2_Click (Value As Integer)
    ' Checks if another queue mode has been specified
    ' if so allows change
    ' Changes values of COST toggle
```

```
    If Queue2.Value = True Then
      QueueCOST = True
    Else
      QueueCOST = False
    End If
End Sub

Sub Queue3_Click (Value As Integer)
    ' Checks if another queue mode has been specified
    ' if so allows change
    ' Changes values of FAST toggle
    If Queue3.Value = True Then
      QueueFAST = True
    Else
      QueueFAST = False
    End If
End Sub

Sub Separ1_Click (Value As Integer)
    ' Accepts only min or max separation
    If Separ1.Value = True Then
      MinSepar = True
    Else
      MinSepar = False
    End If
End Sub

Sub Separ2_Click (Value As Integer)
    ' Accepts only min or max separation
    If Separ2.Value = True Then
      MinSepar = False
    Else
      MinSepar = True
    End If
End Sub

Sub Speed0_Click (Value As Integer)
    ' Checks if another speed has been specified
    ' if so allows change
    If Speed0.Value = True Then
      SpeedOne = True
      If SpeedMulti = False Then
        Combo1.Text = "No"
        Combo1.Enabled = False
      Else
        Combo1.Enabled = True
      End If
    Else
      SpeedOne = False
```

```
      Combo1.Enabled = True
   End If
End Sub

Sub Speed1_Click (Value As Integer)
   ' Checks if another speed has been specified
   ' if so allows change
   If Speed1.Value = True Then
      SpeedMulti = True
      Combo1.Enabled = True
   Else
      SpeedMulti = False
      If SpeedOne = True Then
         Combo1.Text = "No"
         Combo1.Enabled = False
      Else
         Combo1.Enabled = True
      End If
   End If
End Sub
```

REPORT.FRM

```
Function CenterValue (Text$, Ys, LongSpace)
  x = Len(Text$)
  If Ys > x Then
    x = Ys
  End If
  Espacio = Int((LongSpace - x) / 2)
  CenterValue = Space$(Espacio) + Text$ + Space$(LongSpace - x - Espacio)
End Function

Sub Form_Load ()
   Text1.FontBold = False
   Text1.Text = Temp1
End Sub
```

RUNPROCE.FRM

```
Sub Command3D1_Click ()
End
End Sub

Sub Command3D2_Click ()
   Process
End Sub

Sub Label6_Click ()

End Sub

Sub cmdCancel_Click ()
   Unload frmRunProcess
End Sub

Sub cmdRun_Click ()
   Divisor = 0
   If QueueFCFS = True Then
      Divisor = Divisor + 1
   End If
   If QueueCOST = True Then
      Divisor = Divisor + 1
   End If
   If QueueFAST = True Then
      Divisor = Divisor + 1
   End If
   If SpeedOne = True And SpeedMulti Then
      Divisor = Divisor * 2
   End If
   Divisor = Divisor * Convoy
   Contador = 10000 / (Divisor * TotalShip)
   Process
   Unload frmRunProcess
End Sub

Sub Form_Load ()
   frmRunProcess.Top = 100
   frmRunProcess.Left = 2250
   frmRunProcess.Height = 4860
   frmRunProcess.Width = 3855
   OpenBasicData
   gauMonitor.Value = 0
   If Simulation = True Then
      lblTProcess.Caption = "Process: Simulation"
```

```
      lblInitDate.Caption = "Init. Date: " & InitDate
   Else
      lblTProcess.Caption = "Process: Model"
      lblInitDate.Caption = "Init. Date: Not necessary"
   End If
   lblQPrio.Caption = "Q. Priority:"
   If QueueFCFS = True Then
      lblQPrio.Caption = lblQPrio.Caption & " FCFS"
   End If
   If QueueCOST = True Then
      lblQPrio.Caption = lblQPrio.Caption & " COST"
   End If
   If QueueFAST = True Then
      lblQPrio.Caption = lblQPrio.Caption & " FAST"
   End If
   If MinSepar = True Then
      lblSeparDist.Caption = "Separ. Dist.: Minimun"
   Else
      lblSeparDist.Caption = "Separ. Dist.: Maximun"
   End If
   If SpeedOne = True Then
      lblSpeed.Caption = "Speed: One-Speed"
   Else
      lblSpeed.Caption = "Speed: Multi-Speed"
   End If
   lblTimeHorizon.Caption = "Time Horizon: " & TimeHorizon & " days"
   lblConvoy.Caption = "Convoy: " & Convoy
   lblOutput.Caption = "Output: "
   If SendtoScreen = True Then
      lblOutput.Caption = lblOutput.Caption & " Screen"
   End If
   If SendtoPrinter = True Then
      lblOutput.Caption = lblOutput.Caption & " Printer"
   End If
   If SendtoFile = True Then
      lblOutput.Caption = lblOutput.Caption & " File"
   End If
End Sub
```

SELCRIT.FRM

```
Sub cmdCancel_Click ()
   Unload frmSelectCriteria
End Sub

Sub cmdOk_Click ()
  ' sets vars
  ConvoyOpt = chkConvoy
  If optLoc = True Then
    SelectChoice = 1
  ElseIf optLwt = True Then
       SelectChoice = 2
     Else
       SelectChoice = 3
  End If
  SaveBasicData
    Unload frmSelectCriteria
End Sub

Sub Form_Load ()
  frmSelectCriteria.Top = 1250
  frmSelectCriteria.Left = 2500
  frmSelectCriteria.Width = 3855
  frmSelectCriteria.Height = 3060
  ' sets vars
  chkConvoy = ConvoyOpt
  Select Case SelectChoice
  Case Is = 1
      optLoc = True
  Case Is = 2
      optLwt = True
  Case Is = 3
      optLTUT = True
  End Select
End Sub
```

SHOWBD.FRM

```
Dim Shared ControlShip   As Integer


Sub FillControls ()
   ControlShip = 1

   ' Sets variables in screen
   ' according to user's data
   ' Sets channel data
   lblLenChannel.Caption = "Length (m): " + Format(ChannelLength, "######")
   lblCTime.Caption = "Cycle Time(hr): " + Format(CycleTime, "######")
   If OneLane Then
      lblLanes.Caption = "Lanes: One Way"
      lblCTime.Visible = True
   Else
      lblLanes.Caption = "Lanes: Two Way"
      'lblCTime.Visible = False
   End If

   ' Sets ship data
   lblType.Caption = "Type: " + Format$(ControlShip, "##")
   lblDWT.Caption = "DWT: " + Format$(SType(ControlShip).DWT, "######")
   lblSpeed.Caption = "Speed(m/min): " + Format$(SType(ControlShip).Speed, "###.##")
   lblLengthShip.Caption = "Length(m):  " + Format$(SType(ControlShip).Length, "###.
##")
   lblWCost.Caption = "Waiting Cost($/hr): " + Format$(SType(ControlShip).WCost * 60
, "####.##")
End Sub

Sub cmdEdit_Click ()
   Unload frmShowBasicData
   Load frmBasicData
   frmBasicData.Show
End Sub

Sub cmdOk_Click ()
   Unload frmShowBasicData
End Sub

Sub Form_Load ()
   frmShowBasicData.Top = 100
   frmShowBasicData.Left = 2250
   frmShowBasicData.Height = 5220
   frmShowBasicData.Width = 2895
   ' Opens file with basic data
   OpenBasicData
   ' Sets basic controls and buttoms
```

```
   ' ShowBasicData
   FillControls
End Sub

Sub spiShip_SpinDown ()
   ControlShip = ControlShip - 1
   If ControlShip < 1 Then
      ControlShip = 3
   End If
   ' Sets variables in screen
   ' according to user's data
   ' Sets channel data
   lblLenChannel.Caption = "Length (m): " + Format(ChannelLength, "######")
   lblCTime.Caption = "Cycle Time(hr): " + Format(CycleTime, "######")
   If OneLane Then
     lblLanes.Caption = "Lanes: One Way"
     lblCTime.Visible = True
   Else
     lblLanes.Caption = "Lanes: Two Way"
     lblCTime.Visible = False
   End If

   ' Sets ship data
   lblType.Caption = "Type: " + Format$(ControlShip, "##")
   lblDWT.Caption = "DWT: " + Format$(SType(ControlShip).DWT, "######")
   lblSpeed.Caption = "Speed(m/min): " + Format$(SType(ControlShip).Speed, "###.##")
   lblLengthShip.Caption = "Length(m):  " + Format$(SType(ControlShip).Length, "###.
##")
   lblWCost.Caption = "Waiting Cost($/hr): " + Format$(SType(ControlShip).WCost * 60
, "####.##")
End Sub

Sub spiShip_SpinUp ()
    ControlShip = ControlShip + 1
   If ControlShip > 3 Then
      ControlShip = 1
   End If
   ' Sets variables in screen
   ' according to user's data
   ' Sets channel data
   lblLenChannel.Caption = "Length (m): " + Format(ChannelLength, "######")
   lblCTime.Caption = "Cycle Time(hr): " + Format(CycleTime, "######")
   If OneLane Then
     lblLanes.Caption = "Lanes: One Way"
     lblCTime.Visible = True
   Else
     lblLanes.Caption = "Lanes: Two Way"
     lblCTime.Visible = False
   End If
```

```
    ' Sets ship data
    lblType.Caption = "Type: " + Format$(ControlShip, "##")
    lblDWT.Caption = "DWT: " + Format$(SType(ControlShip).DWT, "######")
    lblSpeed.Caption = "Speed(m/min): " + Format$(SType(ControlShip).Speed, "###.##")
    lblLengthShip.Caption = "Length(m):  " + Format$(SType(ControlShip).Length, "###.
##")
    lblWCost.Caption = "Waiting Cost($/hr): " + Format$(SType(ControlShip).WCost * 60
, "####.##")
End Sub
```

USESHPDA.FRM

```
Dim ChangedShip
Dim Control As Integer

Const Color1 = &HFFFFFF      ' White
Const Color2 = &H0           ' Black


Sub ChangeColor ()
  ' Changes color to show marked for deletion record
  ShipArrival.BackColor = Color2
  ShipType.BackColor = Color2
  ArrivalM.BackColor = Color2
  ArrivalD.BackColor = Color2
  ArrivalY.BackColor = Color2
  ArrivalH.BackColor = Color2
  ArrivalMi.BackColor = Color2
End Sub

Sub ChangeColorBack ()
  ' Changes color to show marked for deletion record
  ShipArrival.BackColor = Color1
  ShipType.BackColor = Color1
  ArrivalM.BackColor = Color1
  ArrivalD.BackColor = Color1
  ArrivalY.BackColor = Color1
  ArrivalH.BackColor = Color1
  ArrivalMi.BackColor = Color1
End Sub

Sub ChangeDate (Control As Integer, MonthDate, DayDate, YearDate, HourDate, MinDate)
  TextDate$ = CVDate(Ship(Control).Arrival)
  MonthDate = Left$(TextDate$, 2)
  If Right$(MonthDate, 1) = "/" Then
    MonthDate = "0" + Left$(MonthDate, 1)
    MonthLength = 3
  Else
    MonthLength = 4
  End If
  DayDate = Mid$(TextDate$, MonthLength, 2)
  If Right$(DayDate, 1) = "/" Then
    DayDate = "0" + Left$(DayDate, 1)
    MonthLength = MonthLength + 2
  Else
    MonthLength = MonthLength + 3
  End If
  YearDate = Mid$(TextDate$, MonthLength, 2)
  If Len(TextDate$) < 9 Then
```

```
      HourDate = "00"
      MinDate = "00"
   Else
      AddHour = Right$(TextDate$, 2)
      If UCase$(AddHour) = "AM" Then
         AddHour = 0
      Else
         AddHour = 12
      End If
      TextDate$ = Right$(TextDate$, 11)
      HourDate = Left$(TextDate$, 2)
      MinDate = Left$(Mid$(TextDate$, 4, 2), 2)
   End If
   HourDate = Val(HourDate) + AddHour
End Sub

Sub ChangeDateBack (DMonth, DDay, DYear, DHour, DMin, BackDate)
   ' Finds out number of days since origin
   ' 01/01/92
   DateInput = Str$(DMonth) + "/" + Str$(DDay) + "/" + Str$(DYear)
   BackDate = DateConversion(DateInput)
   ' Finds out decimal part
   BackDate = BackDate + DHour / 24 + DMin / 1440 + 33605
End Sub

Sub ChangeRecord (Control As Integer)
   ' Changes values in input boxes
   ShipType.Text = Ship(Control).SType
   ShipArrival.Text = Ship(Control).AOrder
   Priority.Value = Ship(Control).Priority
   Call ChangeDate(Control, Month1, Day1, Year1, Hour1, Min1)
   ArrivalM.Text = " " + Month1
   ArrivalD.Text = " " + Day1
   ArrivalY.Text = " " + Year1
   ArrivalH.Text = " " + Hour1
   ArrivalMi.Text = " " + Min1
   If MarkForDeletion(Val(ShipArrival)) = True Then
      ChangeColor
   Else
      ChangeColorBack
   End If
End Sub

Sub ClearBoxes ()
   ' Erases text from boxes
   ShipType.Text = " "
   ArrivalM.Text = " "
   ArrivalD.Text = " "
   ArrivalY.Text = " "
```

```
    ArrivalH.Text = " "
    ArrivalMi.Text = " "
    Priority.Value = 0
End Sub

Sub DisableCells ()
    ' Changes Status
    ShipType.Enabled = Not ShipType.Enabled
    ArrivalM.Enabled = Not ArrivalM.Enabled
    ArrivalD.Enabled = Not ArrivalD.Enabled
    ArrivalY.Enabled = Not ArrivalY.Enabled
    ArrivalH.Enabled = Not ArrivalH.Enabled
    ArrivalMi.Enabled = Not ArrivalMi.Enabled
    Priority.Enabled = Not Priority.Enabled
End Sub

Sub NextRec_Click ()
    ' Adds One to Control and changes record
    ' on screen
    Control = Control + 1
    If Control > TotalShip Then
       Control = 1
    End If
    ChangeRecord (Control)
End Sub

Sub PreviousRec_Click ()
    ' Substracts One from Control and changes record
    ' on screen
    Control = Control - 1
    If Control <= 0 Then
       Control = TotalShip
    End If
    ChangeRecord (Control)
End Sub

Sub AddShip_Click ()
    ' Show message
    Message1.Visible = True
    Message1.Caption = "Changes will be effective after Exiting"
    ' Toggles off unnecesary keys
    If EditShip.Enabled = True Then
       ' Main key sequence
       ' Disables edit, delete and printer keys
       EditShip.Enabled = False
       DelShip.Enabled = False
       Printer.Enabled = False
       ' Clears input boxes
       ShipArrival.Text = TotalShip + 1
```

```
      ClearBoxes
      ' Allows for changes in input boxes
      DisableCells
   Else
      ' Saves new vessel
      Open "c:\vb\tesis\ship235.txt" For Append As #1
      Var1 = Val(ShipType.Text)
      Call ChangeDateBack(Val(ArrivalD.Text), Val(ArrivalM.Text), Val(ArrivalY.Text)
, Val(ArrivalH.Text), Val(ArrivalMi.Text), FinalDate)
      Var2 = FinalDate
      Var3 = Priority.Value
      Print #1, Var1, Var2, Var3
      Close #1
      ' Adds new vessel to count and array
      TotalShip = TotalShip + 1
      Ship(TotalShip).AOrder = TotalShip
      Ship(TotalShip).SType = Var1
      Ship(TotalShip).Arrival = Var2
      Ship(TotalShip).Priority = Var3
      ' Clears boxes and waits for new one
      ShipArrival.Text = TotalShip + 1
      ClearBoxes
   End If
End Sub

Sub CancelShip_Click ()
   If Printer.Enabled = True Then
      ' Cancel Key for Ship Menu
      ' Marks record for deletion in DB
      For i = 1 To TotalShip
         If MarkForDeletion(i) = True Then
            Ship(i).Arrival = 0
         End If
      Next i
      ' Orders data base as FCFS,
      ' erases marked records and saves
      OrderFCFS
      ' Goes back to main menu
      Unload frmUserShpData
   Else
      If EditShip.Caption = "Save" Then
         ' Editing menu
         ' Goes back to Ship Menu
         ' Changes caption
         EditShip.Caption = "Edit"
         ' Enable Add and Delete keys
         AddShip.Enabled = True
         Printer.Enabled = True
         DelShip.Enabled = True
```

```
        ' Disables cells for editing
        If ArrivalD.Enabled = True Then
          DisableCells
        End If
      End If
      If EditShip.Enabled = False Then
        ' On some other menu
        If AddShip.Enabled = True Then
          ' Add ship menu
          ' Goes back to Ship Menu
          ' Shows first record in DB
          ChangeRecord (1)
          ' Enables (or disables) necesary keys
          DisableCells
          EditShip.Enabled = True
          Printer.Enabled = True
          DelShip.Enabled = True
        End If
        If DelShip.Enabled = True Then
          ' Enables (or disables) necesary keys
           AddShip.Enabled = True
           EditShip.Enabled = True
           Printer.Enabled = True
        End If
      End If
    End If
End Sub

Sub DelShip_Click ()
    If EditShip.Enabled = True Then
      ' On main menu, disable
      ' unnecesary keys
      AddShip.Enabled = False
      EditShip.Enabled = False
      Printer.Enabled = False
      ' Disable Cells for edition
      If ArrivalD.Enabled = True Then
        DisableCells
      End If
    Else
      If ShipArrival.BackColor <> Color2 Then
        ' Out of main menu,
        ' mark current ship for deletion
        ' if not already marked
        ShipArrival.BackColor = Color2
        ShipType.BackColor = Color2
        ArrivalM.BackColor = Color2
        ArrivalD.BackColor = Color2
        ArrivalY.BackColor = Color2
```

```
        ArrivalH.BackColor = Color2
        ArrivalMi.BackColor = Color2
        MarkForDeletion(Val(ShipArrival)) = True
      Else
        ' if already marked, unmarks
        ShipArrival.BackColor = Color1
        ShipType.BackColor = Color1
        ArrivalM.BackColor = Color1
        ArrivalD.BackColor = Color1
        ArrivalY.BackColor = Color1
        ArrivalH.BackColor = Color1
        ArrivalMi.BackColor = Color1
        MarkForDeletion(Val(ShipArrival)) = False
      End If
    End If
End Sub

Sub EditShip_Click ()
    ' Show message
    ' Show message
    Message1.Visible = True
    Message1.Caption = "Changes will be effective after Exiting"
    If EditShip.Caption = "Edit" Then
      EditShip.Caption = "Save"
      ' Disable Add, Print and Delete keys
      AddShip.Enabled = False
      Printer.Enabled = False
      DelShip.Enabled = False
      ' Enables cells for editing
      DisableCells
      ' Enables (or disables) necesary keys
      EditShip.Enabled = True
    Else
      ' Changes vessel in array
      ' Saves when exiting routine
      CurrentShip = Val(ShipArrival)
      Ship(CurrentShip).SType = Val(ShipType)
      Call ChangeDateBack(Val(ArrivalD.Text), Val(ArrivalM.Text), Val(ArrivalY.Text)
, Val(ArrivalH.Text), Val(ArrivalMi.Text), FinalDate)
      Ship(CurrentShip).Arrival = FinalDate
      Ship(CurrentShip).Priority = Priority.Value
      ' A ship has changed, turns toggle
      ChangedShip = True
    End If
End Sub

Sub Form_Load ()
    ' Calls up basic data
    OpenBasicData
```

```
    ' Calls up ship data
    OpenShipDb
    ' Clears array for record deletion
    For i = 1 To TotalShip
        MarkForDeletion(i) = False
    Next i
    ' Defines basic variables
    Control = 1
    CancelControl = 0
    ' Shows First ship in data base
    ShipType.Text = Ship(Control).SType
    ShipArrival.Text = Ship(Control).AOrder
    Priority.Value = Ship(Control).Priority
    Call ChangeDate(Control, Month1, Day1, Year1, Hour1, Min1)
    ArrivalM.Text = " " + Month1
    ArrivalD.Text = " " + Day1
    ArrivalY.Text = " " + Year1
    ArrivalH.Text = " " + Hour1
    ArrivalMi.Text = " " + Min1
    ' Disables variables
    DisableCells
    ShipDWT.Enabled = False
    ShipSpeed.Enabled = False
    ShipLength.Enabled = False
    ShipWCost.Enabled = False
    ' Enables keys
    EditShip.Enabled = True
    DelShip.Enabled = True
    ' Hides message label
    Message1.Visible = False
End Sub

Sub ShipArrival_LostFocus ()
    Control = ShipArrival.Text
    ShipType.Text = Ship(Control).SType
    ShipArrival.Text = Ship(Control).AOrder
    ShipDWT.Text = "   " + Format$(SType(Ship(Control).SType).DWT, "###,####")
    ShipLength.Text = "   " + Format$(SType(Ship(Control).SType).Length, "###.##")
    ShipSpeed.Text = "   " + Format$(SType(Ship(Control).SType).Speed, "###.##")
    ShipWCost.Text = "   " + Format$(SType(Ship(Control).SType).WCost, "#,###.##")
    Priority.Value = Ship(Control).Priority
    Call ChangeDate(Control, Month1, Day1, Year1, Hour1, Min1)
    ArrivalM.Text = " " + Month1
    ArrivalD.Text = " " + Day1
    ArrivalY.Text = " " + Year1
    ArrivalH.Text = " " + Hour1
    ArrivalMi.Text = " " + Min1
End Sub
```

```
Sub ShipDWT_GotFocus ()
   ' Extend highlight over entire text
   ' entry when user selects input box
   ShipDWT.SelStart = 0
   ShipDWT.SelLength = Len(ShipDWT.Text)
End Sub

Sub ShipDWT_LostFocus ()
   'Read new Ship DWT
   Dim NewShipDWT
   NewShipDWT = Val(ShipDWT.Text)
   ShipDWT.Text = NewShipDWT
   Cr$ = Chr$(13) + Chr$(10)
   Texto1$ = "A ship's dead weight can not be less than zero."
   Texto2$ = "DWT rarely exceeds 500,000 tons." + Cr$
   Texto2$ = Texto2$ + "     Is the entry correct?"
   Texto3$ = "About Ship Tonnage"
   OkDWT = ValidateEntry(NewShipDWT, 0, 500000, Texto1$, Texto2$, Texto3$)
   If OkDWT = True Then
     ShipDWT = NewShipDWT
   End If
End Sub

Sub ShipLength_GotFocus ()
   ' Extend highlight over entire text
   ' entry when user selects input box
   ShipLength.SelStart = 0
   ShipLength.SelLength = Len(ShipLength.Text)
End Sub

Sub ShipLength_LostFocus ()
   'Read new Ship Length
   Dim NewShipLength
   NewShipLength = Val(ShipLength.Text)
   ShipLength.Text = NewShipLength
   Cr$ = Chr$(13) + Chr$(10)
   Texto1$ = "A ship's length can not be less than zero."
   Texto2$ = "A ship's length rarely exceeds 250 m." + Cr$
   Texto2$ = Texto2$ + "        Is the entry correct?"
   Texto3$ = "About Ship Dimensions"
   OkLength = ValidateEntry(NewShipLength, 0, 250, Texto1$, Texto2$, Texto3$)
   If OkLength = True Then
     ShipLength = NewShipLength
   End If
End Sub

Sub ShipSpeed_GotFocus ()
   ' Extend highlight over entire text
   ' entry when user selects input box
```

```
   ShipSpeed.SelStart = 0
   ShipSpeed.SelLength = Len(ShipSpeed.Text)
End Sub

Sub ShipSpeed_LostFocus ()
   'Read new Ship Speed
   Dim NewShipSpeed
   NewShipSpeed = Val(ShipSpeed.Text)
   ShipSpeed.Text = NewShipSpeed
   Cr$ = Chr$(13) + Chr$(10)
   Texto1$ = "A ship's speed can not be less than zero."
   Texto2$ = "A ship's speed rarely exceeds 15 knots." + Cr$
   Texto2$ = Texto2$ + "        Is the entry correct?"
   Texto3$ = "About Ship Speed"
   OkSpeed = ValidateEntry(NewShipSpeed, 0, 15, Texto1$, Texto2$, Texto3$)
   If OkSpeed = True Then
     ShipSpeed = NewShipSpeed
   End If
End Sub

Sub ShipType_Change ()
   ' Checks number input and acts accordingly
   If Val(ShipType.Text) = 1 Or Val(ShipType.Text) = 2 Or Val(ShipType.Text) = 3 Then
     ' Input value ok
     ' Updates ship characteristics on screen
     ShipDWT = "   " + Format(SType(Val(ShipType.Text)).DWT, "###,###")
     ShipLength = "   " + Format(SType(Val(ShipType.Text)).Length, "####")
     ShipSpeed = "   " + Format(SType(Val(ShipType.Text)).Speed, "###.#")
     ShipWCost = "   " + Format(SType(Val(ShipType.Text)).WCost * 60, "###.#")
   Else
     ' Input value not ok
     If Val(ShipType.Text) = 0 Then
       ' A letter or 0 has been input
       ' Clears cells
       ShipDWT = "   "
       ShipLength = "  "
       ShipSpeed = "  "
       ShipWCost = "  "
     Else
       ' A number other than 0,1,2,3 has been input
       ' Changes data back to initial
       CurShip = Val(ShipArrival.Text)
       ShipType.Text = Ship(CurShip).SType
       ShipDWT = "   " + Format(SType(Ship(CurShip).SType).DWT, "###,###")
       ShipLength = "   " + Format(SType(Ship(CurShip).SType).Length, "####")
       ShipSpeed = "   " + Format(SType(Ship(CurShip).SType).Speed, "###.#")
       ShipWCost = "   " + Format(SType(Ship(CurShip).SType).WCost * 60, "###.#")
     End If
   End If
```

```
End Sub

Sub ShipType_LostFocus ()
   Select Case CancelControl
   Case Is = 1
      ShipDWT.Text = "   " + Format$(SType(Ship(Control).SType).DWT, "###,###")
      ShipLength.Text = "  " + Format$(SType(Ship(Control).SType).Length, "###.##")
      ShipSpeed.Text = "  " + Format$(SType(Ship(Control).SType).Speed, "###.##")
      ShipWCost.Text = "  " + Format$(SType(Ship(Control).SType).WCost * 60, "#,###.
##")
   Case Is = 3
      ShipDWT.Text = "   " + Format$(SType(Ship(Control).SType).DWT, "###,###")
      ShipLength.Text = "  " + Format$(SType(Ship(Control).SType).Length, "###.##")
      ShipSpeed.Text = "  " + Format$(SType(Ship(Control).SType).Speed, "###.##")
      ShipWCost.Text = "  " + Format$(SType(Ship(Control).SType).WCost * 60, "#,###.
##")
      Priority.Value = Ship(Control).Priority
      Call ChangeDate(Control, Month1, Day1, Year1, Hour1, Min1)
      ArrivalM.Text = " " + Month1
      ArrivalD.Text = " " + Day1
      ArrivalY.Text = " " + Year1
      ArrivalH.Text = " " + Hour1
      ArrivalMi.Text = " " + Min1
      Ship(Control).SType = Val(ShipType.Text)
   End Select
End Sub

Sub ShipWCost_LostFocus ()
   'Read new Ship WCost
   Dim NewShipWCost
   NewShipWCost = Val(ShipWCost.Text)
   ShipWCost.Text = NewShipWCost
   Cr$ = Chr$(13) + Chr$(10)
   Texto1$ = "A ship's waiting cost can not be less than zero."
   Texto2$ = "Waiting cost rarely exceeds $100/hour." + Cr$
   Texto2$ = Texto2$ + "       Is the entry correct?"
   Texto3$ = "About Waiting Cost"
   OkWCost = ValidateEntry(NewShipWCost, 0, 100, Texto1$, Texto2$, Texto3$)
   If OkWCost = True Then
      ShipWCost = NewShipWCost
   End If
End Sub

Sub spiArrival_SpinDown ()
   Control = Control - 1
   If Control <= 0 Then
      Control = TotalShip
   End If
   ChangeRecord (Control)
```

```
End Sub

Sub spiArrival_SpinUp ()
   ' Adds One to Control and changes record
   ' on screen
   Control = Control + 1
   If Control > TotalShip Then
     Control = 1
   End If
   ChangeRecord (Control)
End Sub
```

MODBAS.BAS

```
Type ShipDataBase
    SType As Integer
    AOrder As Integer
    Arrival As Double
    Priority As Integer
End Type

Type ShipType
    Speed As Single
    DWT As Single
    Length As Single
    WCost As Single
End Type

Type ShipService
    Stime As Double
    Dist As Single
    Speed As Single
End Type

Const NShip = 10000

' Basic Statistical Data
Global ShipTypes

'Global Simulation
Global Ship(NShip) As ShipDataBase
Global ShipT(NShip) As ShipDataBase
Global SType(5) As ShipType
Global NType(3) As Integer
Global Service As Double
Global Pool As Double
Global WaitCost As Double
Global ServiceInd(3)  As Double
Global PoolInd(3)  As Double
Global WaitingC(3)  As Double
Global ServiceTime(NShip)  As Double
Global PoolTime(NShip)  As Double

Global TotalDWT As Double
Global DWTInd(3) As Double

Global MarkForDeletion(1000) As Double
Global Bt(2, 3) As ShipService
Global ChannelTimeOne(3), ChannelTimeMulti(3)
Global ExpectedServiceMulti, ExpectedServiceOne
```

```
' Basic Modeling/Simulation Data
Global Contador, AGauge
Global ChannelLength, OneLane
Global TotalShip  As Integer

Global QueueFCFS, QueueCOST, QueueFAST
Global SpeedMulti, SpeedOne, IniDate, SimDays
Global CycleTime, Convoy, MinSepar
Global MaxSpeed, MaxCost
Global SendtoScreen, SendToPrinter, SendtoFile

Global ConvoyOpt, SelectChoice

Global Ship1, Ship2, Ship3
Global ArrivalRate
Global Temp1

Const Pi = 3.14159


Function AddSpaces (NCharac, Var)
   ' Adds necesary spaces to string
   X = Len(Format$(Var, "#####.#0"))
   AddSpaces = Space$(NCharac - X) + Format$(Var, "#####.#0")
End Function

Sub BasicDataRoutine ()
  ' Shows current channel + procesing data
  Load frmBasicData
  frmBasicData.Show
End Sub

Sub ChangeOrder (Im, Ik)
   ' Switches position of two vessels
   For Im2 = Im To Ik + 1 Step -1
      ShipT(0).SType = ShipT(Im2).SType
      ShipT(0).Priority = ShipT(Im2).Priority
      ShipT(0).Arrival = ShipT(Im2).Arrival
      ShipT(Im2).SType = ShipT(Im2 - 1).SType
      ShipT(Im2).Priority = ShipT(Im2 - 1).Priority
      ShipT(Im2).Arrival = ShipT(Im2 - 1).Arrival
      ShipT(Im2 - 1).SType = ShipT(0).SType
      ShipT(Im2 - 1).Priority = ShipT(0).Priority
      ShipT(Im2 - 1).Arrival = ShipT(0).Arrival
   Next Im2
End Sub

Sub ChangeShipTemp ()
   ' Substitutes array for processing with current array
```

```
   For i = 1 To TotalShip
      ShipT(i).SType = Ship(i).SType
      ShipT(i).Arrival = Ship(i).Arrival
      ShipT(i).Priority = Ship(i).Priority
      ShipT(i).AOrder = Ship(i).AOrder
   Next i
End Sub

Function DateConversion (DateInput)
  ' Changes date in regular format to days
  ' base date: january 1, 1992
  ReDim MonthDays(12)
  MonthDays(1) = 31
  MonthDays(2) = 28
  MonthDays(3) = 31
  MonthDays(4) = 30
  MonthDays(5) = 31
  MonthDays(6) = 30
  MonthDays(7) = 31
  MonthDays(8) = 31
  MonthDays(9) = 30
  MonthDays(10) = 31
  MonthDays(11) = 30
  MonthDays(12) = 31
  MonthDate = Val(Left$(DateInput, 2))
  DayDate = Val(Mid$(DateInput, 3, 2))
  YearDate = Val(Right$(DateInput, 2))
  Days = 0
  Extra = 0
  If YearDate > 92 Then
    ExtraDays = YearDate - 92
    Extras = Int(ExtraDays / 4)
    Days = ExtraDays * 365 + Extras + Days + 1
  End If
  For i = 1 To MonthDate - 1
    Days = Days + MonthDays(i)
  Next i
  Days = Days + (DayDate - 1) + HourDate / 24 + MinDate / 1440
  DateConversion = Days
End Function

Sub ErrorTrap ()
End Sub

Function ExpectedService (SpeedMode)
   ' Determines mean inter-vessel arrival
   MeanInterval = 0
   For i = 2 To TotalShip
     ArrivalInterval = Ship(i).Arrival - Ship(i - 1).Arrival
```

```
    MeanInterval = MeanInterval + ArrivalInterval
  Next i
  MeanInterval = MeanInterval / (TotalShip - 1)
  ' Determines minimum speed and mean length
  MinSpeed = 10 ^ 16
  For i = 1 To 3
    ExpLength = NType(i) / TotalShip * SType(i).Length
    If SType(i).Speed < MinSpeed Then
      MinSpeed = SType(i).Speed
    End If
  Next i
  ' Determines minimum speed separation
  ' for one speed (speedmode=0) and multi speed (speedmode=1)
  If MinSpeed Then
    ExService = ExpLength / MinSpeed * (.094 * MinSpeed ^ .75 + 2#)
  Else
    ExService = ExpLength / MinSpeed * (.168 * MinSpeed ^ .75 + 2.8)
  End If
  If SpeedMode = 0 Then ' one speed
    ExpectedService = ExService
  Else                  ' multi speed
    ExpSum = 0
    For i = 1 To 3
      For j = 1 To 3
        Call SeparationMulti(i, j, SType(i).Speed, Mintime, MinDist)
        Temp = 1 / SType(i).Speed - 1 / SType(j).Speed
        If Temp > 0 Then
          ExpSum = (ChannelLength - MinDist) * Temp * NType(i) / TotalShip * NTy
pe(j) / TotalShip + ExpSum
        End If
      Next j
    Next i
  End If

End Function

Sub FinalResult ()
  ' Computes final results for
  ' Mean service and waiting time
  Service = 0: Pool = 0
  For Ik = 1 To TotalShip
    Service = Service + ServiceTime(Ik)
    Pool = Pool + PoolTime(Ik)
  Next Ik
  Service = Service / TotalShip
  Pool = Pool / TotalShip
  ' Computes mean service and waiting time per ship type
  For Ik = 1 To 3
    ServiceInd(Ik) = ServiceInd(Ik) / NType(Ik)
```

```
      PoolInd(Ik) = PoolInd(Ik) / NType(Ik)
      WaitingC(Ik) = SType(Ik).WCost * PoolInd(Ik)
      DWTInd(Ik) = DWTInd(Ik) * SType(Ik).DWT
   Next Ik
   TotalDWT = 0
   For Ik = 1 To 3
      WaitCost = NType(Ik) / TotalShip * WaitingC(Ik) + WaitCost
      TotalDWT = DWTInd(Ik) + TotalDWT
   Next Ik
End Sub

Sub GenerateArrival ()
  ' SimDay in days
  ' ArrivalRate in minutes
  Randomize 1
  ' Changes date format
  'BaseDate = DateConversion(IniDate)
  ' Checks if previous file exists
   ' Checks if Generated ship data base exits
   Xfile = Dir$("C:\simmod\ShipGen.txt")
   If UCase(X) <> UCase("ShipGen.txt") Then
     ' File does no exist
     ' Generates new
     FileCopy "c:\simmod\blank.txt", "c:\simmod\ShipGen.txt"
   Else
     ' File exists
     ' Renames *.bak
     FileCopy "c:\simmod\ShipGen.txt", "c:\simmod\ship.bak"
     FileCopy "c:\simmod\blank.Txt", "c:\simmod\ShipGen.txt"
   End If
  ' Opens file to include data
  Open "c:\simmod\ShipGen.TXT" For Output As #1
  ' Determines number of ships to generate
  TotalShip = Int(SimDays * 24 / ArrivalRate)
  ' Determines steps in progress bar
  ProgresStep = 3000 / TotalShip
  NStep = Int(TotalShip / 50)
  ' Runs generation routine
  i = 0
  L = Exp(-ArrivalRate)
  M = 0
  XAcum = 0
  Do
   X = 0
   p = 1
   ok = True
   ' Finds arrival with Poisson Distribution
   Do
     u = Rnd
```

```
    p = p * u
    If p > L Then
      X = X + 1
    Else
      Exit Do
    End If
  Loop While ok
  XAcum = XAcum + X
  ' Finds Ship with Uniform Distribution
  Y = Rnd
  If Y < Ship1 / 100 Then
    Tship = 1
  ElseIf Y < (Ship1 + Ship2) / 100 Then
      Tship = 2
    Else
      Tship = 3
  End If
  ' Prints data into file
  Print #1, Tship, XAcum, 0
  M = M + 1
 Loop While M < TotalShip
 Close #1
End Sub

Sub GenShpArrival ()
  ' Loads form for automatic generation
  ' of arrivals
  Load frmGenShpData
  frmGenShpData.Show
End Sub

Sub LineIntersect (X1, Y1, M1, X2, Y2, M2, Xres, Yres)
  ' Finds intersection point of two lines
  ' X represents time, Y represents distance
  ' M represents speed
  B1 = Y1 - M1 * X1
  B2 = Y2 - M2 * X2
  Xres = (B2 - B1) / (M1 - M2)
  Yres = (M1 * Xres + B1)
End Sub 'LineIntersect

Sub MultiSpeed (k, PoolM, SpeedMode)
  ' Checks is minimun separation is conserved
  ' otherwise assigns minimum separation time to
  ' arrival time - Finds Pool time
  Call SeparationMulti(ShipT(k).SType, ShipT(k - 1).SType, Bt(1, 0).Speed, Mintime,
MinDist)
  If SType(ShipT(k).SType).Speed > Bt(1, 0).Speed Then
    ' Next ship faster than current
```

```
      If Mintime + Bt(1, 0).Stime < ShipT(k).Arrival Then
        ' Ship arrives after minimum time - no wait
        Bt(2, 0).Stime = ShipT(k).Arrival
        PoolM = 0
      Else
        ' Ship arrives before minimum time - wait
        Bt(2, 0).Stime = Mintime + Bt(1, 0).Stime
        PoolM = Bt(2, 0).Stime - ShipT(k).Arrival
      End If
   Else
      ' Next ship slower than current
      Ttemp = SType(ShipT(k).SType).Speed / (Bt(1, 1).Dist - MinDist)
      If Bt(1, 1).Stime - Ttemp < ShipT(k).Arrival Then
        ' Minimum distance ok to enter - no wait
        Bt(2, 0).Stime = ShipT(k).Arrival
        PoolM = 0
      Else
        ' Minimum distance not ok - wait
        Bt(2, 0).Stime = Bt(1, 1).Stime - Ttemp
        PoolM = Bt(2, 0).Stime - ShipT(k).Arrival
      End If
   End If
End Sub ' ProcessMulti

Sub OneSpeed (k, PoolO)
   ' Checks is minimun separation is conserved
   ' otherwise assigns minimum separation time to
   ' arrival time
   Call SeparationOne(ShipT(k).SType, ShipT(k - 1).SType, Bt(1, 0).Speed, Mintime, M
inDist)
   ' rem
   If Mintime + Bt(1, 0).Stime < ShipT(k).Arrival Then
      ' Mininum distance ok to enter - no wait
      Ttemp = ShipT(k).Arrival
      PoolO = 0
   Else
      ' Minimum distance not ok - wait
      Ttemp = Mintime + Bt(1, 0).Stime
      PoolO = Bt(1, 0).Stime + Mintime - ShipT(k).Arrival
   End If
End Sub

Sub OpenBasicData ()
   On Error GoTo ErrorTrap
   ' Opens data base with channel data + reads it
   ' True=-1; False=0
   Handler = 1
   Open "c:\simmod\Basic.dat" For Input As #1
   Input #1, Simulation, SendtoScreen, SendToPrinter, SendtoFile
```

```
    Input #1, ChannelLength, OneLane, CycleTime
    Input #1, Var1
    Input #1, Var2
    Input #1, Var3
    Input #1, Var5
    Input #1, Var6
    Input #1, Var7
    Input #1, Var8
    QueueFCFS = Val(Var1)
    QueueCOST = Val(Var2)
    QueueFAST = Val(Var3)
    SpeedMulti2 = Val(Var4)
    SpeedMulti = Val(Var5)
    SpeedOne = Val(Var6)
    Convoy = Val(Var7)
    MinSepar = Val(Var8)
    Input #1, SimDays
    ' Inputs ships characteristics
    Input #1, ShipTypes
    MaxSpeed = -10 ^ 6
    MaxCost = -10 ^ 6
    For i = 1 To ShipTypes
        Input #1, Var1, Var2, Var3, Var4
        SType(i).Speed = Val(Var1)
        SType(i).Length = Val(Var2)
        SType(i).DWT = Val(Var3)
        SType(i).WCost = Val(Var4)
        If MaxSpeed < Val(Var1) Then
           MaxSpeed = Val(Var1)
        If MaxCost < Val(Var4) Then
           MaxCost = Val(Var4)
        End If
        End If
    Next i
    ' Inputs simulation data
    ' used only if user specifies
    Input #1, Var1, Var2, Var3
    Input #1, Var4
    Ship1 = Val(Var1)
    Ship2 = Val(Var2)
    Ship3 = Val(Var3)
    ArrivalRate = Val(Var4)
    Input #1, Var1, Var2
    ConvoyOpt = Var1
    SelectChoice = Var2
    Close #1
    ' Sets begining of simulation to current date
    IniDate = Date
GoTo Jump2
```

```
ErrorTrap:
   Trampa = Err
   If Trampa = 62 Then
      msg = "The current Basic Data file is corrupted or missing."
      msg = msg & "  Press Ok to correct this problem or Cancel for no corrections.
"
      Answer = MsgBox(msg, 49, "File Error")
      If Answer = 1 Then
         Close
         FileCopy "c:\simmod\auxiliar.dat", "c:\simmod\Basic.dat"
      End If
      Resume OPenBasicData2
OPenBasicData2:
      Open "c:\simmod\Basic.dat" For Input As #1
      Input #1, Simulation, SendtoScreen, SendToPrinter, SendtoFile
      Input #1, ChannelLength, OneLane, CycleTime
      Input #1, Var1
      Input #1, Var2
      Input #1, Var3
      Input #1, Var5
      Input #1, Var6
      Input #1, Var7
      Input #1, Var8
      QueueFCFS = Val(Var1)
      QueueCOST = Val(Var2)
      QueueFAST = Val(Var3)
      SpeedMulti2 = Val(Var4)
      SpeedMulti = Val(Var5)
      SpeedOne = Val(Var6)
      Convoy = Val(Var7)
      MinSepar = Val(Var8)
      Input #1, SimDays
      ' Inputs ships characteristics
      Input #1, ShipTypes
      MaxSpeed = -10 ^ 6
      MaxCost = -10 ^ 6
      For i = 1 To ShipTypes
         Input #1, Var1, Var2, Var3, Var4
         SType(i).Speed = Val(Var1)
         SType(i).Length = Val(Var2)
         SType(i).DWT = Val(Var3)
         SType(i).WCost = Val(Var4)
         If MaxSpeed < Val(Var1) Then
            MaxSpeed = Val(Var1)
            If MaxCost < Val(Var4) Then
               MaxCost = Val(Var4)
            End If
         End If
      Next i
```

```
    ' Inputs simulation data
    ' used only if user specifies
        Input #1, Var1, Var2, Var3
        Input #1, Var4
        Ship1 = Val(Var1)
        Ship2 = Val(Var2)
        Ship3 = Val(Var3)
        ArrivalRate = Val(Var4)
        Input #1, Var1, Var2
        ConvoyOpt = Var1
        SelectChoice = Var2
        Close #1
    ' Sets begining of simulation to current date
        IniDate = Date
    End If

Jump2:
End Sub

Sub OpenShipDb ()
    ' Opens file containing ship data base
    ' and determines number of ships
    ' The file to use depends on user's specs
    If Simulation = True Then
       Open "C:\simmod\Ship235.txt" For Input As #1
    Else
       Open "C:\simmod\ShipGen.TXT" For Input As #1
    End If
    i = 1
    NType(1) = 0
    NType(2) = 0
    NType(3) = 0
    Do While Not EOF(1)
      Input #1, Ship(i).SType
      Input #1, Ship(i).Arrival
      Input #1, Ship(i).Priority
      Ship(i).AOrder = i
      Select Case Ship(i).SType
      Case Is = 1
          NType(1) = NType(1) + 1
      Case Is = 2
          NType(2) = NType(2) + 1
      Case Is = 3
          NType(3) = NType(3) + 1
      End Select
      i = i + 1
    Loop
    TotalShip = i - 1
    Close #1
```

```
  Call ChangeShipTemp
End Sub

Sub OrderFCFS ()
  ' Orders db based on arriving date
  Dim Tempo As ShipDataBase
  ReDim MatrixP(TotalShip + 1, 5) As Variant
  For i = 1 To TotalShip
     MatrixP(i, 1) = Ship(i).SType
     MatrixP(i, 2) = Ship(i).Arrival
     MatrixP(i, 3) = Ship(i).Priority
  Next i
  ' Orders DB by Arrival Date [MatrixP(i,2)]
  Nx = 0
  Do While Nx <= TotalShip
    Nx = Nx + 1
    For Order = 1 To TotalShip
       If (MatrixP(Order + 1, 2) < MatrixP(Order, 2)) And (MatrixP(Order + 1, 2) <
> 0) Then
          Tempo.SType = MatrixP(Order, 1)
          Tempo.Arrival = MatrixP(Order, 2)
          Tempo.Priority = MatrixP(Order, 3)
          MatrixP(Order, 1) = MatrixP(Order + 1, 1)
          MatrixP(Order, 2) = MatrixP(Order + 1, 2)
          MatrixP(Order, 3) = MatrixP(Order + 1, 3)
          MatrixP(Order + 1, 1) = Tempo.SType
          MatrixP(Order + 1, 2) = Tempo.Arrival
          MatrixP(Order + 1, 3) = Tempo.Priority
       End If
    Next Order
  Loop
  ' Saves file ordered by arrival date
  FileName$ = "c:\simmod\Ship235.txt"
  Open FileName$ For Output As #2
  Counter = 0
  For i = 1 To TotalShip
     If MatrixP(i, 2) <> 0 Then
       ' Saves only vessels not marked for deletion
       ' (Ship(i).Arrival=0)
       Counter = Counter + 1
       Print #2, MatrixP(i, 1);
       Print #2, MatrixP(i, 2);
       Print #2, MatrixP(i, 3)
     End If
  Next i
  Close #2
  TotalShip = Counter
  ' Sets ships by ordered arrival date
  For i = 1 To TotalShip
```

```
      Ship(i).SType = MatrixP(i, 1)
      Ship(i).Arrival = MatrixP(i, 2)
      Ship(i).Priority = MatrixP(i, 3)
      Ship(i).AOrder = i
   Next i
   Call ChangeShipTemp
End Sub 'OrderFCFS

Sub PrincipalMenu ()
    'Returns user to main menu
    Load PrincipMenu
    PrincipMenu.Show
End Sub

Sub Process ()
    ' Call all data
    OpenBasicData
    OpenShipDb
    ' Changes Mouse pointer to hourglass
    PrincipMenu.MousePointer = 11
    ' Determines service time for one and
    ' multi speed modes
    ExpectedServiceOne = ExpectedService(0)
    ExpectedServiceMulti = ExpectedService(1)
    ' If printer control is on prints headline
    If SendToPrinter Then
      Printer.FontBold = True
      Printer.FontSize = 9.75
      Printer.Print "TRAFFIC CONTROL MODEL"
      Printer.Print
      If Simulation Then
        Printer.Print "SIMULATION RESULTS"
      Else
        Printer.Print "MODEL RESULTS"
      End If
      Printer.FontSize = 8.25
    End If
    ' Calculates time in channel
    Call SystemTime
    ' Sets up procesing for FCFS
    If QueueFCFS Then
      ' Single vessel convoys
      If SpeedOne Then
        ProcessSingle 1, 0
        If SendtoFile = True Then
          SaveResults 1, 0, 1
        End If
        If SendtoScreen = True Then
          ShowResults 1, 0, 1
```

```
      End If
      If SendToPrinter = True Then
        Impre 1, 0, 1
      End If
    End If
    If SpeedMulti Then
      ProcessSingle 1, 1
      If SendtoFile = True Then
        SaveResults 1, 1, 1
      End If
      If SendtoScreen = True Then
        ShowResults 1, 1, 1
      End If
      If SendToPrinter = True Then
        Impre 1, 1, 1
      End If
    End If
    If Convoy > 1 Then
      For i = 2 To Convoy
        If SpeedMulti Then
          ProcessConvoy 1, 1, i
          If SendtoFile = True Then
            SaveResults 1, 1, i
          End If
          If SendtoScreen = True Then
            ShowResults 1, 1, i
          End If
          If SendToPrinter = True Then
            Impre 1, 1, i
          End If
        End If
      Next i
    End If
  End If
  ' Sets up procesing for FAST
  If QueueFAST Then
    ' Single vessel convoys
    If SpeedMulti Then
      ProcessSingle 2, 1
      If SendtoFile = True Then
        SaveResults 2, 1, 1
      End If
      If SendtoScreen = True Then
        ShowResults 2, 1, 1
      End If
      If SendToPrinter = True Then
        Impre 2, 1, 1
      End If
    End If
```

```
    If Convoy > 1 Then
      For i = 2 To Convoy
        If SpeedMulti Then
          ProcessConvoy 2, 1, i
          If SendtoFile = True Then
            SaveResults 2, 1, i
          End If
          If SendtoScreen = True Then
            ShowResults 2, 1, i
          End If
          If SendToPrinter = True Then
            Impre 2, 1, i
          End If
        End If
      Next i
    End If
  End If
  ' Sets up procesing for COST
  If QueueCOST Then
    ' Single vessel convoys
    If SpeedOne Then
      ProcessSingle 3, 0
      If SendtoFile = True Then
        SaveResults 3, 0, 1
      End If
      If SendtoScreen = True Then
        ShowResults 3, 0, 1
      End If
      If SendToPrinter = True Then
        Impre 3, 0, 1
      End If
    End If
    If SpeedMulti Then
      ProcessSingle 3, 1
      If SendtoFile = True Then
        SaveResults 3, 1, 1
      End If
      If SendtoScreen = True Then
        ShowResults 3, 1, 1
      End If
      If SendToPrinter = True Then
        Impre 3, 1, 1
      End If
    End If
    If Convoy > 1 Then
      For i = 2 To Convoy
        If SpeedMulti Then
          ProcessConvoy 3, 1, i
          If SendtoFile = True Then
```

```
              SaveResults 3, 1, i
            End If
            If SendtoScreen = True Then
              ShowResults 3, 1, i
            End If
            If SendToPrinter = True Then
              Impre 3, 1, i
            End If
          End If
       Next i
     End If
   End If
   ' Opens report form
   If SendtoScreen = True Then
      Load Report
      Report.Show
   End If
   ' Changes mouse pointer back to arrow
   PrincipMenu.MousePointer = 0
   ' Ejects printer page
   If SendToPrinter = True Then
     Printer.EndDoc
   End If
End Sub

Sub ProcessConvoy (Queue, SpeedMode, ConvoyLength)
   ConvoyBase = 1
   ' Defines first ship
   Bt(1, 0).Stime = ShipT(1).Arrival
   Bt(1, 0).Dist = 0
   Bt(1, 0).Speed = SType(ShipT(1).SType).Speed
   Bt(1, 1).Speed = SType(ShipT(1).SType).Speed
   Bt(1, 1).Stime = ChannelLength / Bt(1, 0).Speed / 60 + Bt(1, 0).Stime
   Bt(1, 1).Dist = ChannelLength
   ServiceTime(1) = 0
   PoolTime(1) = 0
   ' Finds group of ships in waiting pool
   For Ik = 2 To TotalShip
     ok = False
     ControlType = 3
     Im2 = TotalShip
     Do
        Call SeparationMulti(ControlType, ShipT(Ik - 1).SType, SType(ControlType).S
peed, Mintime, MinDist)
        If ControlType > Ship(Ik - 1).SType Then
          Maxtime = (ChannelLength / SType(ShipT(Ik - 1).SType).Speed - (ChannelLe
ngth - MinDist) / SType(ControlType).Speed) / 60
        Else
          Maxtime = Mintime
```

```
        End If
        For Im = Ik + 1 To Im2
            If Bt(1, 0).Stime + Maxtime < ShipT(Im).Arrival Then
                ImFinal = Im - 1
                Im = Im2
            End If
            If Ship(Im).SType = ControlType Then
                ok = True
            Else
                ok = False
            End If
        Next Im
        If ok = True Then
            Exit Do
        Else
            ControlType = ControlType - 1
            If ControlType <= 0 Then Exit Do
        End If
    Loop While ok = False
    ServiceTime(Ik) = Mintime
    ' Organizes vessels in pool according to user's criteria
    ' but only if vessel in ready spot has no priority
    OkChanges = 0
    If ShipT(Ik).Priority <> 1 Then
        ' Finds if any ship in pool has priority
        For Im = Ik + 1 To ImFinal
            If ShipT(Im).Priority = 1 Then
                Call ChangeOrder(Im, Ik)
                Im = ImFinal
                OkChanges = 1
            End If
        Next Im
        ' If no ship in pool has priority, organizes
        ' acording to convoy length.
        ' If convoy lenght maximum or no vessels
        ' available then selects by queue criteria
        If OkChanges <> 1 Then
            ChangeInConvoy = 0
            If ConvoyBase < ConvoyLength Then
                ' Finds if any ship in pool can complete convoy
                For Im = Ik + 1 To ImFinal
                    If ShipT(Ik).SType = ShipT(Im).SType Then
                        Call ChangeOrder(Im, Ik)
                        Im = ImFinal
                        ConvoyBase = ConvoyBase + 1
                        ChangeInConvoy = 1
                    End If
                Next Im
            End If
```

```
        If ConvoyBase = ConvoyLength And ChangeInConvoy <> 1 Then
          ' No vessels to add to convoy has been found or
          ' convoy is already full.
          ' Next vessel assigned according to queueing criteria
          Select Case Queue
          Case Is = 2 'FAST
            If SType(ShipT(Ik).SType).Speed <> MaxSpeed Then
              For Im = Ik + 1 To ImFinal
                Select Case SType(ShipT(Ik).SType).Speed
                Case Is = MaxSpeed
                  Call ChangeOrder(Im, Ik)
                  Im = ImFinal
                Case Is < SType(ShipT(Im).SType).Speed
                  Call ChangeOrder(Im, Ik)
                End Select
              Next Im
            End If
          Case Is = 3 'COST
            If SType(ShipT(Ik).SType).WCost <> MaxCost Then
              For Im = Ik + 1 To ImFinal
                Select Case SType(ShipT(Ik).SType).WCost
                Case Is = MaxCost
                  Call ChangeOrder(Im, Ik)
                  Im = ImFinal
                Case Is < SType(ShipT(Im).SType).WCost
                  Call ChangeOrder(Im, Ik)
                End Select
              Next Im
            End If
          End Select
          ConvoyBase = 1
        End If' New convoy
      End If'No other ships has priority
    End If ' Ship in ready spot has no priority
    ' Finds pool time
    Select Case SpeedMode
    Case Is = 1
      Call MultiSpeed(Ik, PoolW, SpeedMode)
    Case Is = 2
    End Select
    PoolTime(Ik) = PoolW * 60
    ServiceInd(ShipT(Ik).SType) = ServiceInd(ShipT(Ik).SType) + ServiceTime(Ik)
    PoolInd(ShipT(Ik).SType) = PoolInd(ShipT(Ik).SType) + PoolTime(Ik)
    DWTInd(ShipT(Ik).SType) = DWTInd(ShipT(Ik).SType) + 1
    ' Changes gauge with process progress
    AGauge = AGauge + Contador
    frmRunProcess.gauMonitor.Value = AGauge
  Next Ik
  ' Restores array to initial state
```

```
    Call ChangeShipTemp
End Sub

Sub ProcessSingle (Queue, SpeedMode)
    ' If One speed, sets speed to lower speed
    If SpeedMode = 0 Then
      MinSpeed = 10 ^ 6
      For j = 1 To 3
          If SType(j).Speed < MinSpeed Then
            MinSpeed = SType(j).Speed
          End If
      Next j
    End If
    ' Defines first ship
    Bt(1, 0).Stime = ShipT(1).Arrival
    Bt(1, 0).Dist = 0
    Select Case SpeedMode
    Case Is = 0 'One speed
        Bt(1, 0).Speed = MinSpeed
        Bt(1, 1).Speed = MinSpeed
    Case Else 'Multi speed
        Bt(1, 0).Speed = SType(ShipT(1).SType).Speed
        Bt(1, 1).Speed = SType(ShipT(1).SType).Speed
    End Select
    Bt(1, 1).Stime = ChannelLength / Bt(1, 0).Speed / 60 + Bt(1, 0).Stime
    Bt(1, 1).Dist = ChannelLength
    ServiceTime(1) = 0
    PoolTime(1) = 0
    ' Finds group of ships in waiting pool
    For Ik = 2 To TotalShip
      Select Case SpeedMode
      Case Is = 0 'One speed
        ControlType = 1
        Im2 = TotalShip
        Call SeparationOne(ControlType, ShipT(Ik - 1).SType, SType(ControlType).Speed
, Mintime, MinDist)
        For Im = Ik + 1 To Im2
            If Bt(1, 0).Stime + Mintime < ShipT(Im).Arrival Then
              ImFinal = Im - 1
              Im = Im2
            End If
        Next Im
      Case Is = 1 'Multi speed
        ok = False
        ControlType = 3
        Im2 = TotalShip
        Do
          Call SeparationMulti(ControlType, ShipT(Ik - 1).SType, SType(ControlType).S
peed, Mintime, MinDist)
```

```
        If ControlType > Ship(Ik - 1).SType Then
          Maxtime = (ChannelLength / SType(ShipT(Ik - 1).SType).Speed - (ChannelLe
ngth - MinDist) / SType(ControlType).Speed) / 60
        Else
          Maxtime = Mintime
        End If
        For Im = Ik + 1 To Im2
          If Bt(1, 0).Stime + Maxtime < ShipT(Im).Arrival Then
            ImFinal = Im - 1
            Im = Im2
          End If
          If Ship(Im).SType = ControlType Then
            ok = True
          Else
            ok = False
          End If
        Next Im
        If ok = True Then
          Exit Do
        Else
          ControlType = ControlType - 1
          If ControlType <= 0 Then Exit Do
        End If
      Loop While ok = False
    End Select
    ServiceTime(Ik) = Mintime
    ' Organizes vessels in pool according to user's criteria
    ' but only if vessel in ready spot has no priority
    OkChanges = 0
    If ShipT(Ik).Priority <> 1 Then
      ' Finds if any ship in pool has priority
      For Im = Ik + 1 To ImFinal
        If ShipT(Im).Priority = 1 Then
          Call ChangeOrder(Im, Ik)
          Im = ImFinal
          OkChanges = 1
        End If
      Next Im
      ' If no ship in pool has priority
      ' organizes acording to queue criteria
      If OkChanges <> 1 Then
        Select Case Queue
        Case Is = 2 'FAST
          If SType(ShipT(Ik).SType).Speed <> MaxSpeed Then
            For Im = Ik + 1 To ImFinal
              Select Case SType(ShipT(Ik).SType).Speed
              Case Is = MaxSpeed
                Call ChangeOrder(Im, Ik)
                Im = ImFinal
```

```
              Case Is < SType(ShipT(Im).SType).Speed
                Call ChangeOrder(Im, Ik)
              End Select
            Next Im
          End If
        Case Is = 3 'COST
          If SType(ShipT(Ik).SType).WCost <> MaxCost Then
            For Im = Ik + 1 To ImFinal
              Select Case SType(ShipT(Ik).SType).WCost
              Case Is = MaxCost
                Call ChangeOrder(Im, Ik)
                Im = ImFinal
              Case Is < SType(ShipT(Im).SType).WCost
                Call ChangeOrder(Im, Ik)
              End Select
            Next Im
          End If
        End Select
      End If ' No Ship has priority
    End If ' Ship in ready spot has no priority
    ' Finds pool time
    Select Case SpeedMode
    Case Is = 0
      Call OneSpeed(Ik, PoolW)
    Case Else
      Call MultiSpeed(Ik, PoolW, SpeedMode)
    End Select
    PoolTime(Ik) = PoolW * 60
    ServiceInd(ShipT(Ik).SType) = ServiceInd(ShipT(Ik).SType) + ServiceTime(Ik)
    PoolInd(ShipT(Ik).SType) = PoolInd(ShipT(Ik).SType) + PoolTime(Ik)
    DWTInd(ShipT(Ik).SType) = DWTInd(ShipT(Ik).SType) + 1
    ' Changes gauge with process progress
    AGauge = AGauge + Contador
    frmRunProcess.gauMonitor.Value = AGauge
  Next Ik
  ' Restores array to initial state
  Call ChangeShipTemp
End Sub

Sub SaveBasicData ()
  ' Opens data base with channel data
  Open "c:\simmod\Basic.dat" For Output As #1
  Print #1, Simulation, SendtoScreen, SendToPrinter, SendtoFile
  Print #1, ChannelLength, OneLane, CycleTime
  Print #1, QueueFCFS
  Print #1, QueueFAST
  Print #1, QueueCOST
  Print #1, SpeedMulti
  Print #1, SpeedOne
```

```
    Print #1, Convoy
    Print #1, MinSepar
    Print #1, SimDays
    Print #1, ShipTypes
    For i = 1 To ShipTypes
       Print #1, SType(i).Speed, SType(i).Length, SType(i).DWT, SType(i).WCost
    Next i
    Print #1, Ship1, Ship2, Ship3
    Print #1, ArrivalRate
    Print #1, ConvoyOpt
    Print #1, SelectChoice
    Close #1
End Sub

Sub SaveResults (Queue, SpeedMode, ConvoyLength)
    ' Calculates final stats
    Call FinalResult
    ' Checks if Results file exists
    Xfile = Dir$("C:\simmod\result.Txt")
    If UCase(Xfile) <> UCase("Result.txt.Dat") Then
      ' File does no exist
      ' Generates new
      FileCopy "c:\simmod\blank.txt", "c:\simmod\Result.txt"
    Else
      ' File exists
      ' Rename *.bak
      FileCopy "c:\simmod\result.txt", "c:\simmod\result.bak"
      FileCopy "c:\simmod\blank.txt", "c:\simmod\result.txt"
    End If
    ' Opens file
    If FiletoSave <> "" Then
       Open FiletoSave For Append As #1
    Else
       Open "c:\simmod\result.txt" For Append As #1
    End If
    ' Print headings and basic info
    Select Case Queue
    Case Is = 1 'FCFS
        PrintQueue = "FCFS"
    Case Is = 2 'FAST
        PrintQueue = "FAST"
    Case Is = 3 'COST
        PrintQueue = "COST"
    End Select
    Select Case SpeedMode
    Case Is = 0
        PrintSpeed = "One-Speed     "
    Case Is = 1
        PrintSpeed = "Multi-Speed  "
```

```
    End Select
    ' Select units
    If Service > 10 Then
      ' Prints results in hours
      TFactor = 1
      RUnit = " [hour]    "
    Else
      ' Prints results in minutes
      TFactor = 60
      RUnit = " [min]     "
    End If
    Select Case MinSepar
    Case True
        PrintSepar = "Minimum Separation"
    Case Else
        PrintSepar = "Maximum Separation"
    End Select
    Print #1, "Queue Routine: "; PrintQueue
    Print #1, "Speed Mode:    "; PrintSpeed;
    Print #1, Space(5); "Separation Distance: "; PrintSepar
    Print #1, "Convoy Length: "; ConvoyLength
    Print #1, Space(27); "TOTAL"; Space(5); "SHIP1"; Space(5); "SHIP2"; Space(5); "SHIP3"
    ' Print results
    Print #1, "Service Time" + RUnit;
    Print #1, AddSpaces(10, Service * TFactor);
    For Ik = 1 To 3
       Print #1, AddSpaces(10, ServiceInd(Ik) * TFactor);
    Next Ik
    Print #1,
    Print #1, "Waiting Time" + RUnit;
    Print #1, AddSpaces(10, Pool * TFactor);
    For Ik = 1 To 3
       Print #1, AddSpaces(10, PoolInd(Ik) * TFactor);
    Next Ik
    Print #1,
    Print #1, "Waiting Cost [$/hr]    ";
    Print #1, AddSpaces(10, WaitCost);
    For Ik = 1 To 3
       Print #1, AddSpaces(10, WaitingC(Ik));
    Next Ik
    Print #1,
    Print #1, "ThroughPut [Mill Ton/hr] ";
    Print #1, TotalDWT / 10 ^ 6;
    Print #1,
    Print #1,
    Close #1
End Sub

Sub SeparationMulti (M, N, ControlSpeed, SeparationTime, SeparationDistance)
```

```
   ' Calculates separation distance and time
   ' Separation Time as defined in thesis
   ' m=Current ship type, n=preceding ship type
   ' Control speed is set by previous ship
   If MinSepar Then
     SeparationDistance = (.14 * ControlSpeed ^ .75 + 1.5) * SType(N).Length
   Else
     SeparationDistance = (.168 * ControlSpeed ^ .75 + 1.8) * SType(N).Length
   End If
   If 1 / SType(M).Speed - 1 / SType(N).Speed > 0 Then
     SeparationTime = (ChannelLength - SeparationDistance) * (1 / SType(ShipT(M).ST
ype).Speed - 1 / SType(N).Speed)
   Else
     SeparationTime = (SeparationDistance + SType(M).Length) / ControlSpeed
   End If
   SeparationTime = SeparationTime / 60
End Sub

Sub SeparationOne (M, N, ControlSpeed, SeparationTime, SeparationDistance)
   ' Calculates separation distance and time
   ' Separation Time as defined in thesis
   ' m=Current ship, n=preceding ship
   ' Control speed is set by previous ship
   If MinSepar Then
     SeparationDistance = (.14 * ControlSpeed ^ .75 + 1.5) * SType(N).Length
   Else
     SeparationDistance = (.168 * ControlSpeed ^ .75 + 1.8) * SType(N).Length
   End If
   SeparationTime = (SeparationDistance + SType(M).Length) / ControlSpeed
   SeparationTime = SeparationTime / 60
End Sub

Sub SetUpSimulation ()
   Load frmProcessData
   frmProcessData.Show
End Sub

Sub ShowResults (Queue, SpeedMode, ConvoyLength)
   ' Calculates final stats
   Call FinalResult
   ' Print headings and basic info
   Select Case Queue
   Case Is = 1 'FCFS
      PrintQueue = "FCFS"
   Case Is = 2 'FAST
      PrintQueue = "FAST"
   Case Is = 3 'COST
      PrintQueue = "COST"
   End Select
```

```
Select Case SpeedMode
Case Is = 0
    PrintSpeed = "One-Speed    "
Case Is = 1
    PrintSpeed = "Multi-Speed  "
End Select
' Select units
If Service > 10 Then
  ' Prints results in hours
  TFactor = 1
  RUnit = " [hour]   "
Else
  ' Prints results in minutes
  TFactor = 60
  RUnit = " [min]    "
End If
Select Case MinSepar
Case True
    PrintSepar = "Minimum Separation"
Case Else
    PrintSepar = "Maximum Separation"
End Select
' Fills text var with headings
Cr$ = Chr$(13) + Chr$(10)
Temp1 = Temp1 + "Queue Routine: " + PrintQueue + Cr$
Temp1 = Temp1 + "Speed Mode:    " + PrintSpeed
Temp1 = Temp1 + Space(5) + "Separation Distance: " + PrintSepar + Cr$
Temp1 = Temp1 + "Convoy Length: " + Str$(ConvoyLength) + Cr$ + Cr$
Temp1 = Temp1 + Space(42) + "TOTAL" + Space(7) + "SHIP1" + Space(7) + "SHIP2" + S
pace(7) + "SHIP3" + Cr$
' Print results
Temp1 = Temp1 + "Service Time" + RUnit + Space(3)
Temp1 = Temp1 + AddSpaces(12, Service * TFactor)
For Ik = 1 To 3
   Temp1 = Temp1 + AddSpaces(15, ServiceInd(Ik) * TFactor)
Next Ik
Temp1 = Temp1 + Cr$
Temp1 = Temp1 + "Waiting Time" + RUnit + Space(3)
Temp1 = Temp1 + AddSpaces(14, Pool * TFactor)
For Ik = 1 To 3
   Temp1 = Temp1 + AddSpaces(15, PoolInd(Ik) * TFactor)
Next Ik
Temp1 = Temp1 + Cr$
Temp1 = Temp1 + "Waiting Cost [$/hr]   " + Space(3)
Temp1 = Temp1 + AddSpaces(14, WaitCost)
For Ik = 1 To 3
   Temp1 = Temp1 + AddSpaces(15, WaitingC(Ik))
Next Ik
Temp1 = Temp1 + Cr$
```

```
   Temp1 = Temp1 + "ThroughPut [Mill Ton/hr]" + Space(3)
   Temp1 = Temp1 + Str$(TotalDWT / 10 ^ 6)
   Temp1 = Temp1 + Cr$ + Cr$
End Sub

Sub SystemTime ()
   ' Calculates travel time in channel
   If SpeedOne Then
      MinSpeed = 10 ^ 30
      For i = 1 To 3
         If SType(i).Speed < MinSpeed Then
            MinSpeed = SType(i).Speed
         End If
      Next i
      For i = 1 To 3
         ChannelTimeOne(i) = ChannelLength / MinSpeed
      Next i
   Else
      For i = 1 To 3
         ChannelTimeMulti(i) = ChannelLength / SType(i).Speed
      Next i
   End If
End Sub

Sub UseShpArrival ()
   ' Loads form for automatic generation
   ' of arrivals
   Load frmUserShpData
   frmUserShpData.Show
End Sub

Function ValidateEntry (Entry, min, Max, TextMin$, TextMax$, TextMsg$)
      ' Checks if Entry less than Minimum
      If Entry <= min Then
         MsgBox TextMin$, 15, TextMsg$
         ValidateEntry = False
      End If
      ' Or larger than Maximum
      If Entry > Max Then
         If MsgBox(TextMax$, 4, TextMsg$) = 6 Then
            ValidateEntry = True
         Else
            ValidateEntry = False
         End If
      End If
      ' Or OK
      If Entry > min And Entry < Max Then
         ValidateEntry = True
      End If
```

End Function

PRINT.BAS

```
Sub Impre (Queue, SpeedMode, ConvoyLength)
   ' Computes final results for
   ' Mean service and waiting time
   Service = 0: Pool = 0
   For Ik = 1 To TotalShip
      Service = Service + ServiceTime(Ik)
      Pool = Pool + PoolTime(Ik)
   Next Ik
   Service = Service / TotalShip
   Pool = Pool / TotalShip
   ' Computes mean service and waiting time per ship type
   For Ik = 1 To 3
      ServiceInd(Ik) = ServiceInd(Ik) / NType(Ik)
      PoolInd(Ik) = PoolInd(Ik) / NType(Ik)
      WaitingC(Ik) = SType(Ik).WCost * PoolInd(Ik)
      DWTInd(Ik) = DWTInd(Ik) * SType(Ik).DWT
   Next Ik
   TotalDWT = 0
   For Ik = 1 To 3
      WaitCost = NType(Ik) / TotalShip * WaitingC(Ik) + WaitCost
      TotalDWT = DWTInd(Ik) + TotalDWT
   Next Ik
   Select Case Queue
   Case Is = 1 'FCFS
       PrintQueue = "FCFS"
   Case Is = 2 'FAST
       PrintQueue = "FAST"
   Case Is = 3 'COST
       PrintQueue = "COST"
   End Select
   Select Case SpeedMode
   Case Is = 0
       PrintSpeed = "One-Speed    "
   Case Is = 1
       PrintSpeed = "Multi-Speed  "
   End Select
   ' Select units
   If Service > 10 Then
     ' Prints results in hours
     TFactor = 1
     RUnit = " [hour]    "
   Else
     ' Prints results in minutes
     TFactor = 60
     RUnit = " [min]     "
```

```
    End If
    Select Case MinSepar
    Case True
        PrintSepar = "Minimum Separation"
    Case Else
        PrintSepar = "Maximum Separation"
    End Select
    Printer.FontBold = False
    Printer.FontSize = 8.25
    Printer.Print "Queue Routine: "; PrintQueue
    Printer.Print "Speed Mode:    "; PrintSpeed;
    Printer.Print Space(5); "Separation Distance: "; PrintSepar
    Printer.Print "Convoy Length: "; ConvoyLength
    Printer.Print Space(27); "TOTAL"; Space(5); "SHIP1"; Space(5); "SHIP2"; Space(5); "SHIP3"
    ' Print results
    Printer.Print "Service Time" + RUnit;
    Printer.Print AddSpaces(10, Service * TFactor);
    For Ik = 1 To 3
       Printer.Print AddSpaces(10, ServiceInd(Ik) * TFactor);
    Next Ik
    Printer.Print
    Printer.Print "Waiting Time" + RUnit;
    Printer.Print AddSpaces(10, Pool * TFactor);
    For Ik = 1 To 3
       Printer.Print AddSpaces(10, PoolInd(Ik) * TFactor);
    Next Ik
    Printer.Print
    Printer.Print "Waiting Cost [$/hr]    ";
    Printer.Print AddSpaces(10, WaitCost);
    For Ik = 1 To 3
       Printer.Print AddSpaces(10, WaitingC(Ik));
    Next Ik
    Printer.Print
    Printer.Print "ThroughPut [Mill Ton/hr] ";
    Printer.Print TotalDWT / 10 ^ 6
    Printer.Print
    Printer.Print
End Sub

Sub Prueba ()
Printer.Print "Prueba de Impresora"
Printer.EndDoc
End Sub
```

APPENDIX B

USER'S MANUAL

In the following pages, a brief description of the operations optimization model is made. The figures are taken directly from the program. Since the program is written with a Windows interface, it can only be run from that operating system. On the other hand, this allows all command functions as are available under that system.

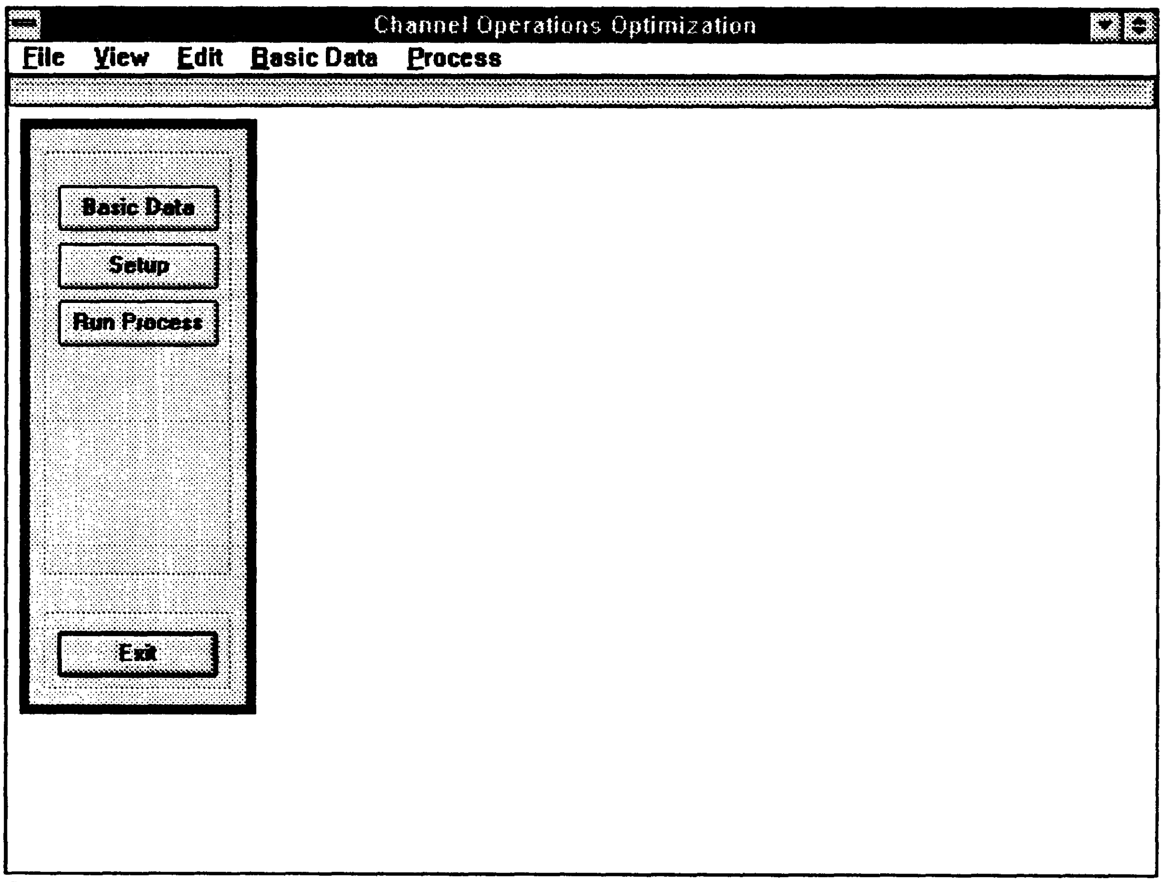


Figure 1

The first figure (Figure No.1) shows the opening screen. The user can access the options through the menu on top or through the buttons on the screen. The corner options allow to close the application, change to another while running or to minimize the screen size as in any Windows application.

The *FILE* command on the top menu allows access to filing and printing functions

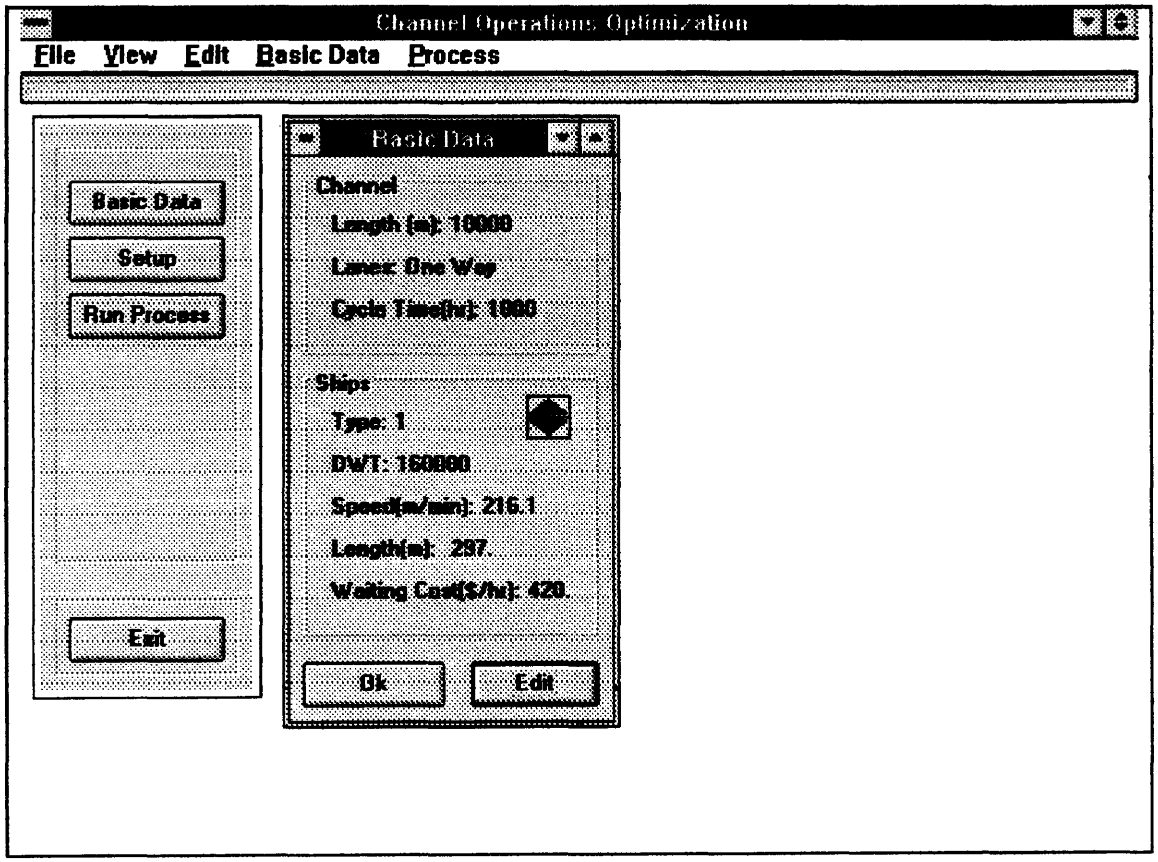


Figure 2

and to exit the program. The *VIEW* command shows the basic channel and ship characteristics as defined by the default (See figure No. 2). Once the user has changed information, the program will maintain that as default.

The *EDIT* command will access routines to change the channel or vessel set up included as basic data (and shown in the *VIEW* command). The *BASIC DATA* command allows access to ship arrival data bases, either user generated or generated by the model. On the first case, (user generated data) the user will have entered a subroutine that allows changes in that data base (Figure 4). On the second case, the user will enter a routine that allows for the generation of arrival with a Poisson distribution (Figure 3).

Finally, *PROCESS* enters the basic processing routine. In this module, the user specifies the combination of factors he wishes to model/simulate and runs it.

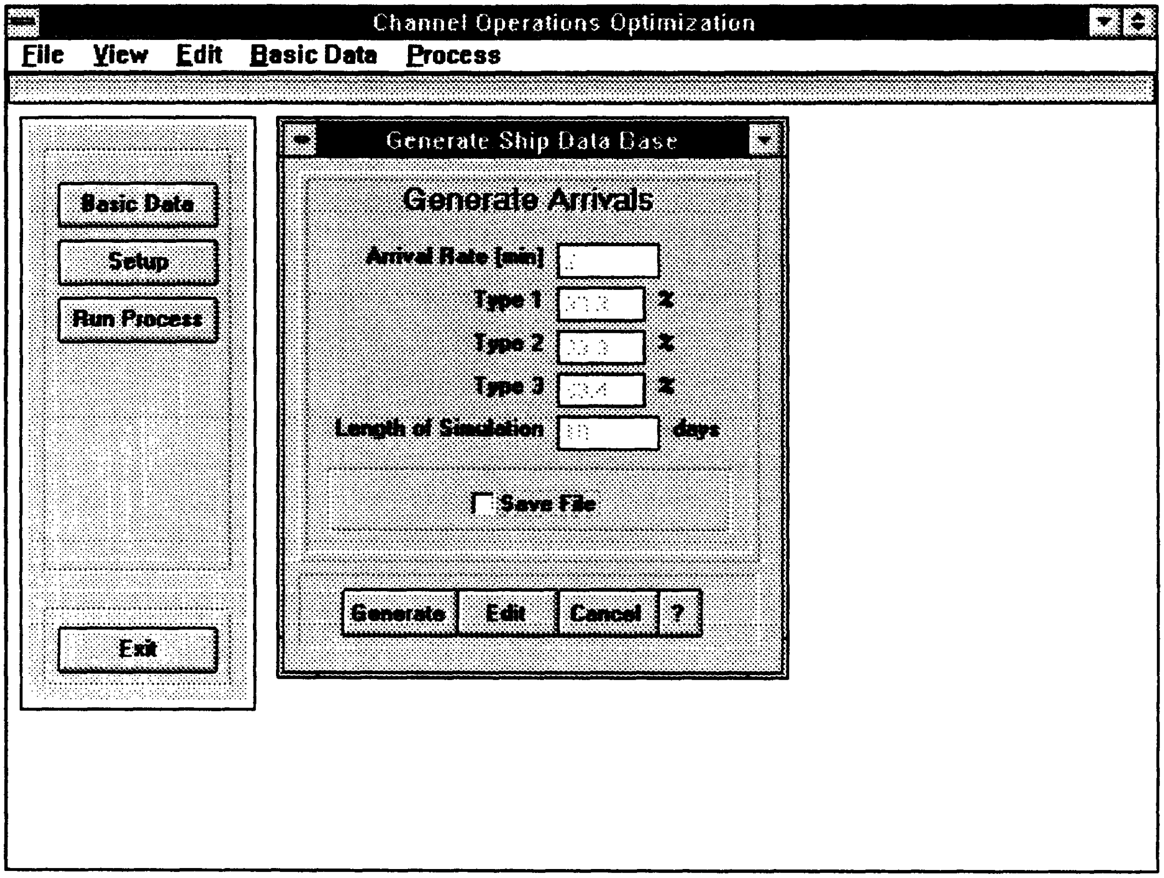


Figure 3

As mentioned before, the user can either use the main menu or access routine through the buttons located to the left of the screen. The first button (from top to bottom) is the *BASIC DATA* button. Selecting this option, brings up the "Basic Data" menu (Figure 4). In this menu, the user sees the channel basic data and the different vessels characteristics. He can then proceed to edit them by pressing the *EDIT* button. This action will bring up a screen as the one shown in figure 5.

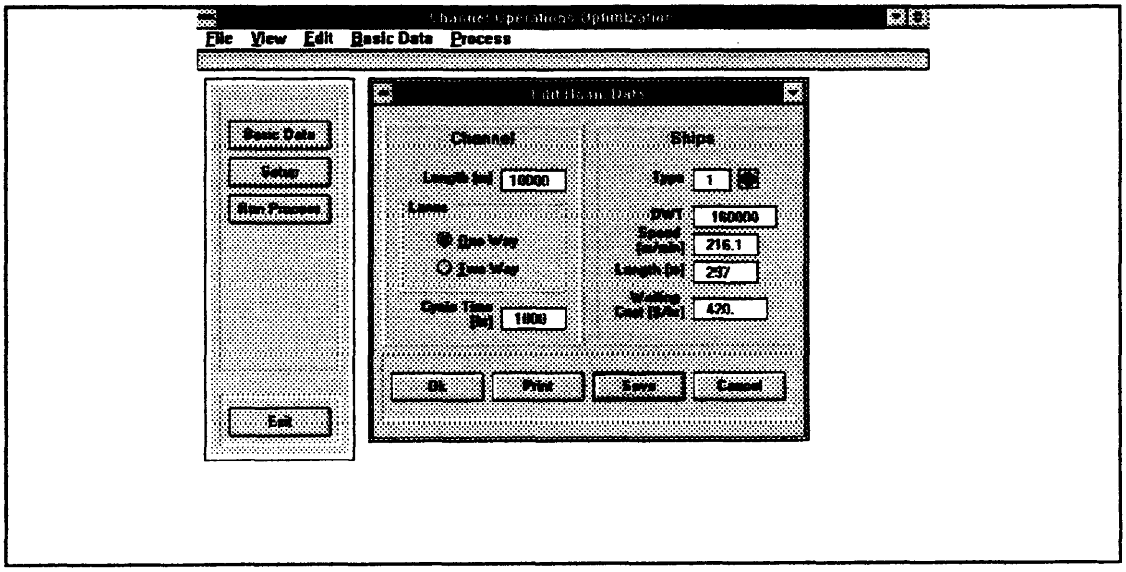


Figure 4

The *EDIT* button in the "Basic Data" menu (Figure 4) allows access to the channel and basic vessel types characteristics. The user can either print the data (*PRINT* key), save changes (*SAVE* key) or go back to the main menu (by pressing *CANCEL*). As for all menus, this screen can be enabled/disabled or minimized by pressing the corner controls. The channel length, the number of lanes, and the cycle time (the number of hours the traffic is inbound or outbound) are defined/modified in this screen. The vessel basic characteristics are also defined. The user must specify three vessel types and their tonnage, speed in the channel, length and waiting cost. The input data will serve as default as soon as the user presses the *SAVE* button.

Another key control in the main menu, the *SETUP* key, will change the parameters for the simulation. Pressing this buttom button bring up a screen as the one shown in figure 5.

This screen is the "Process Set-Up" menu. The user defines the parameters for the simulationn or modeling. The user models when he has an arrival data base generated by the program (in the screen shown on figure 4). Otherwise, the user can simulate with a ship arrival data base (as shown in figure 3).

The specifications for speed mode, separation distance, queue priority and use of convoys are all set up in this screen. The user just places the cursor on top of the

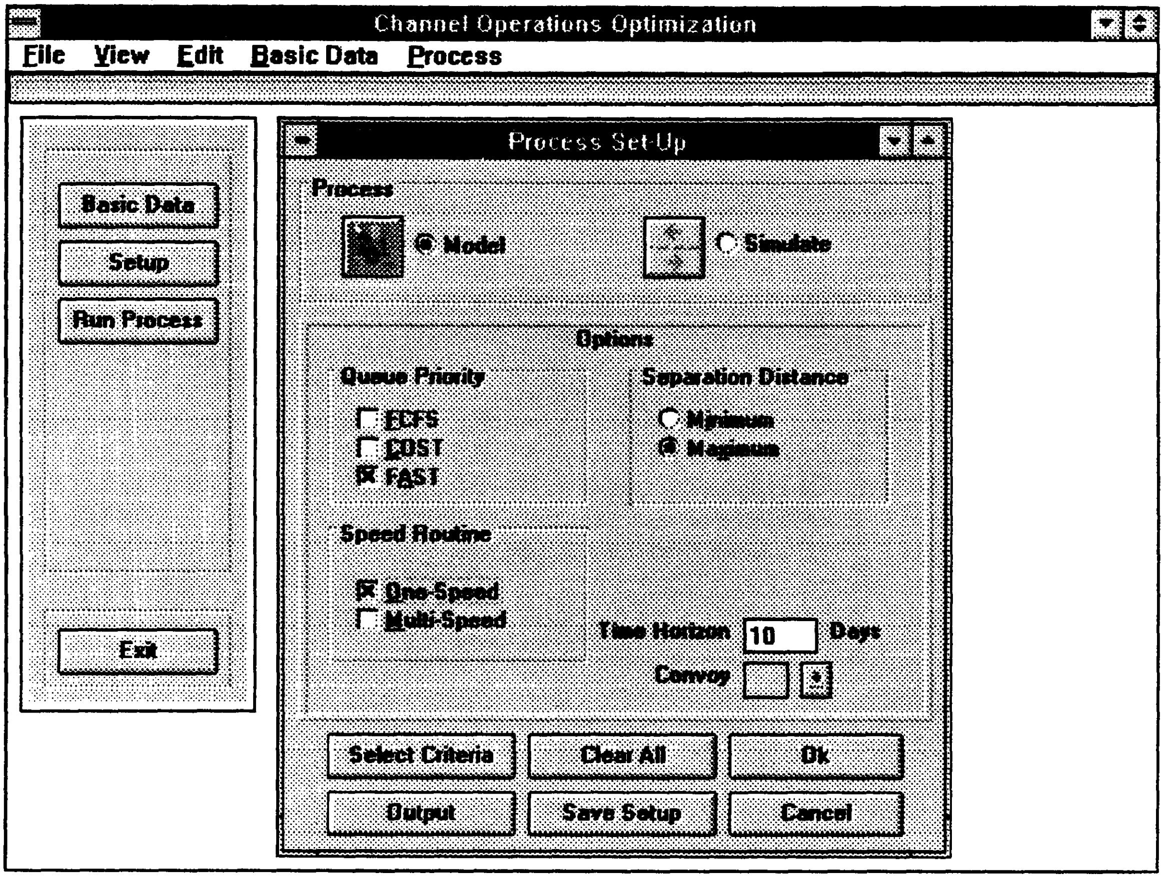


Figure 5

option and changes the toggle on/off for most options. The "Time Horizon" option allows the user to define the time period starting on the current date for which the simulation/model is run.
The *OUTPUT* button changes the way the results are printed. They final tables can be either sent to the screen, to a printer or to a file. The way the options are presented is shown in figure 6.

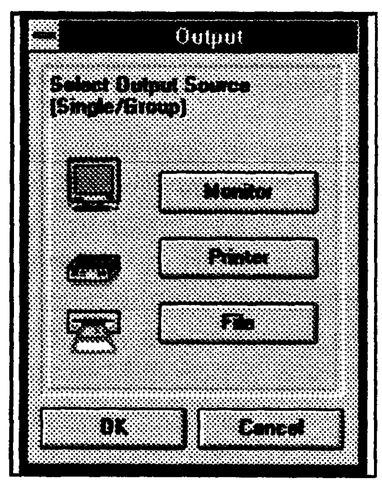


Figure 6

The user can also request a criteria for the model to choose a "best" solution or combination of factors. By pressing the *SELECT CRITERIA* button the user has access to the screen shown in figure 7.

Once the user has indicated the variables he wishes to model, he can save them pressing the *SAVE SET-UP* key. The model will not

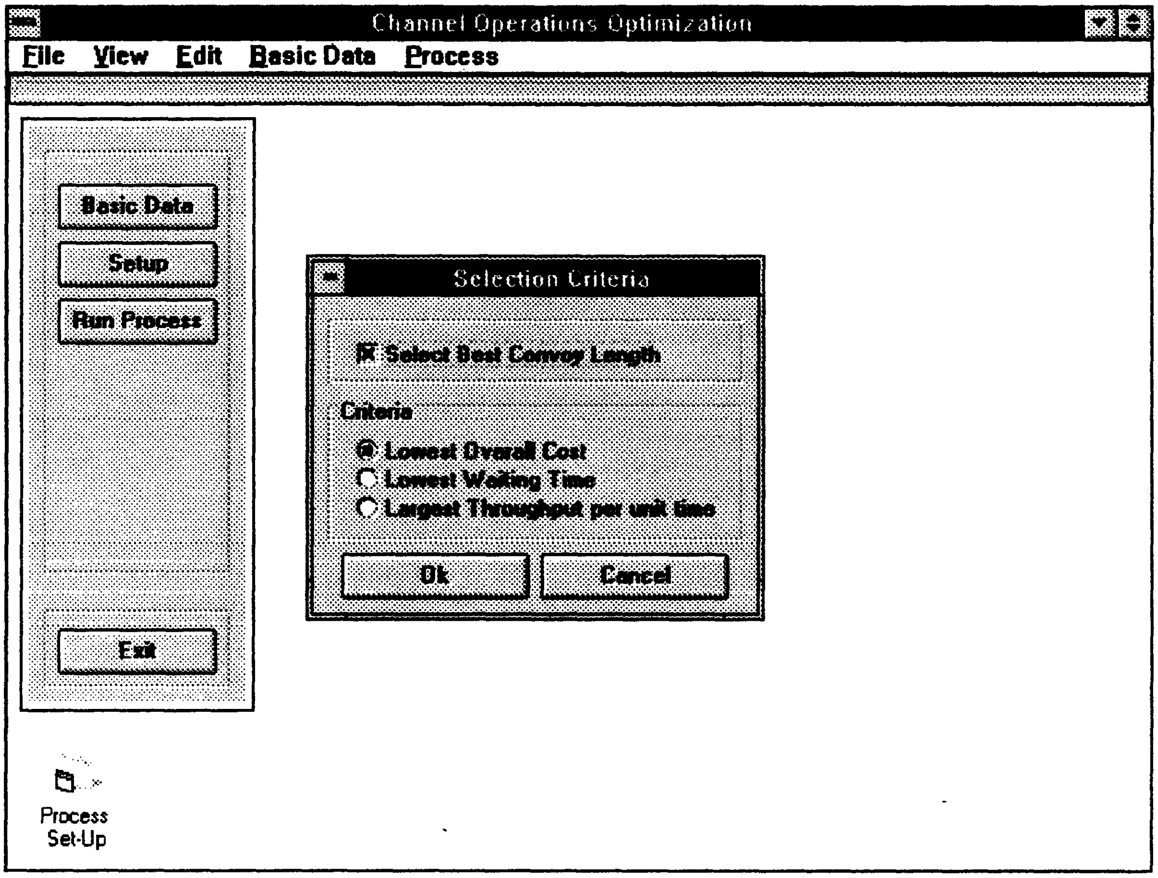


Figure 7

recognize any changes unless the user presses the previous

The user is now ready to run a simulation/model. In order to do this, he must press the *RUN PROCESS* key on the main menu (Figure 1). This action will call up a screen similar to the one shown in Figure 8. The "Run Process" screen shows in short the options that will be considered when running. The lower part of the screen contains a bar to control the time it will take the model to come up with results. This will obviously be a function of the simulation time or number of vessels. Once the bar reaches 100%, the program will display the results in the screen (Figure 9) if the user has requested that option when on "Process Set-Up" menu. Otherwise, the program will send the output to the device requested.

Finally, in the main menu, the *EXIT* key will end the program.

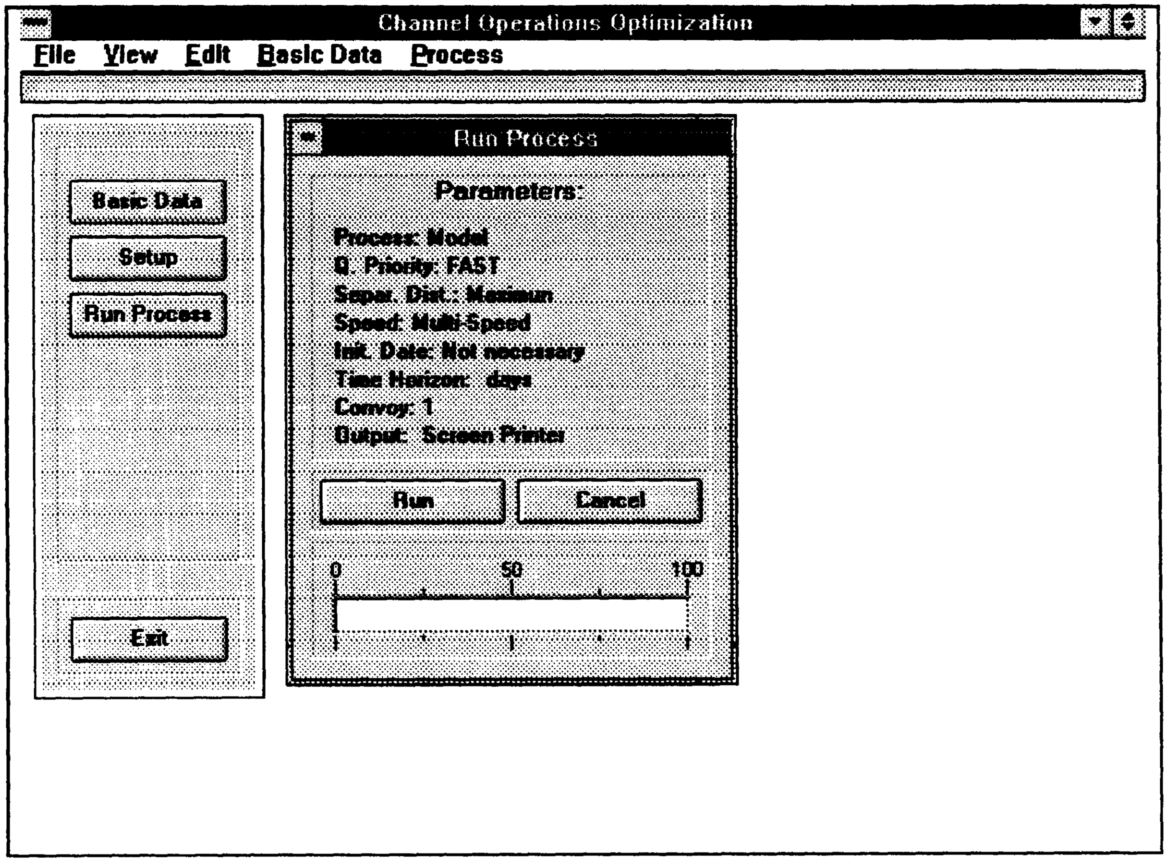

Channel Operations Optimization
File View Edit Basic Data Process
Basic Data
Setup
Run Process
Exit
Run Process
Parameters:
Process: Model
Q. Priority: FAST
Separ. Dist.:
Speed: Multi-Speed
Init. Date: Not necessary
Time Horizon: days
Convoy: 1
Output: Screen Printer
Run
Cancel
0
50
100

Figure 8

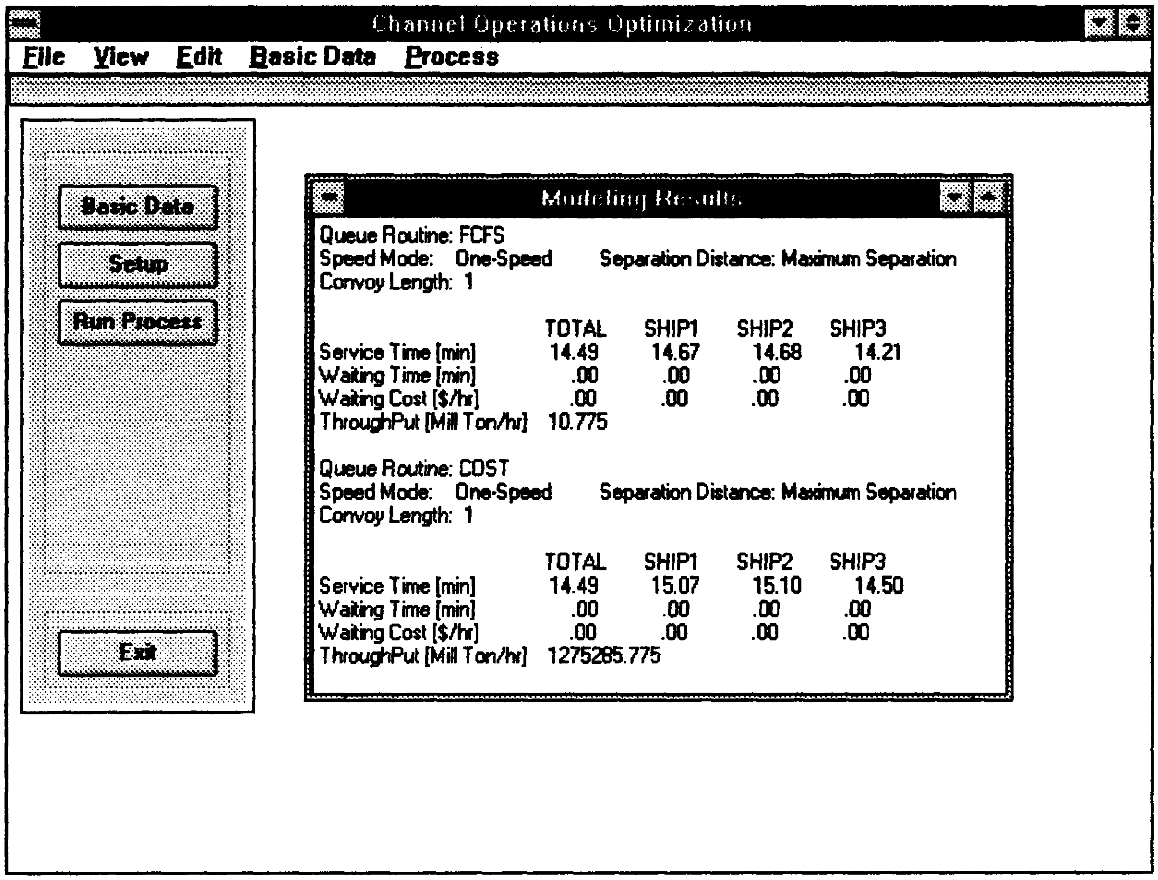

Channel Operations Optimization
File View Edit Basic Data Process
Basic Data
Setup
Run Process
Exit
Modeling Results
Queue Routine: FCFS
Speed Mode: One-Speed
Separation Distance: Maximum Separation
Convoy Length: 1
TOTAL SHIP1 SHIP2 SHIP3
Service Time [min] 14.49 14.67 14.68 14.21
Waiting Time [min] .00 .00 .00 .00
Waiting Cost [$/hr] .00 .00 .00 .00
ThroughPut [Mill Ton/hr] 10.775
Queue Routine: COST
Speed Mode: One-Speed
Separation Distance: Maximum Separation
Convoy Length: 1
TOTAL SHIP1 SHIP2 SHIP3
Service Time [min] 14.49 15.07 15.10 14.50
Waiting Time [min] .00 .00 .00 .00
Waiting Cost [$/hr] .00 .00 .00 .00
ThroughPut [Mill Ton/hr] 1275285.775

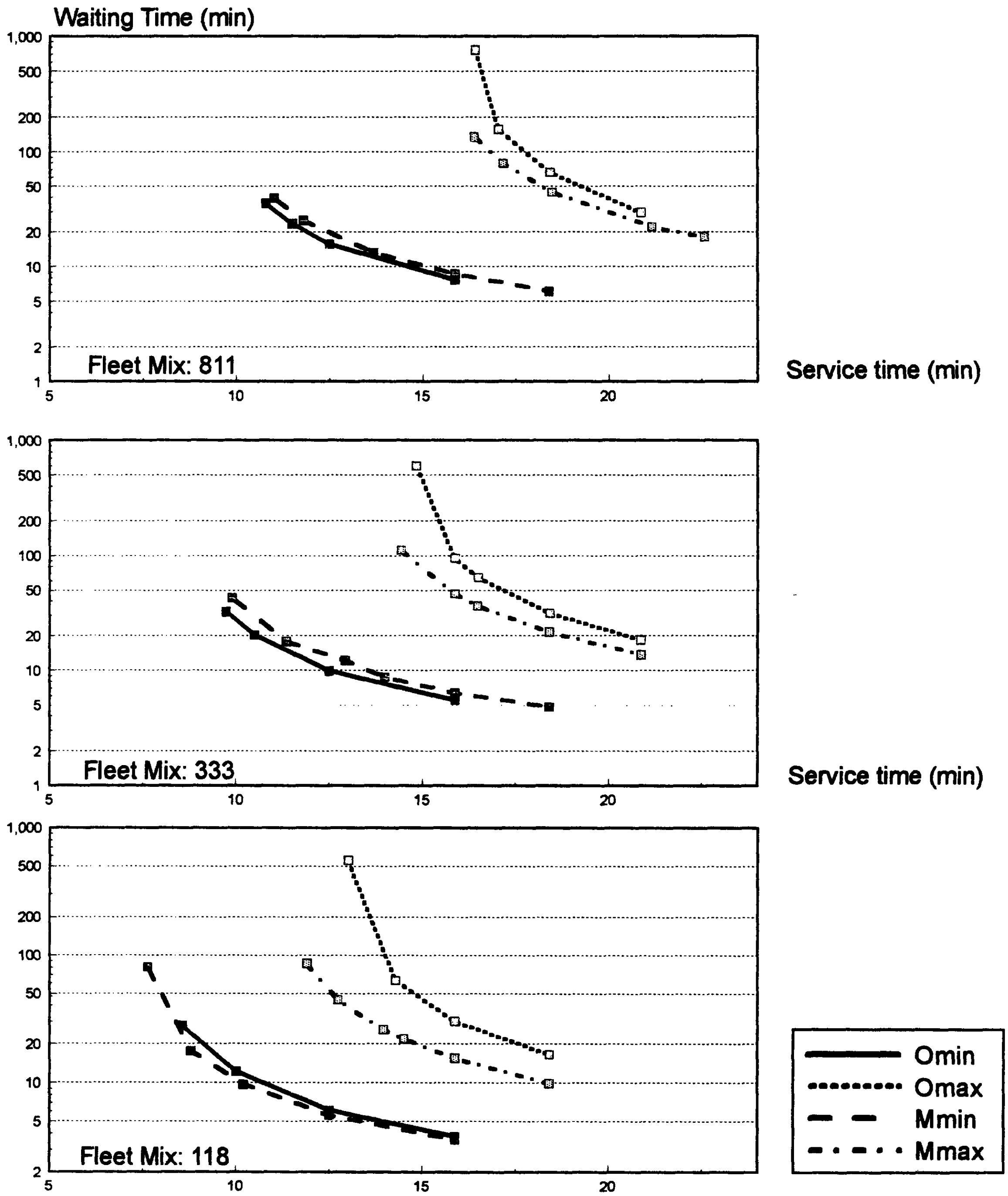
Figure 9

APPENDIX C

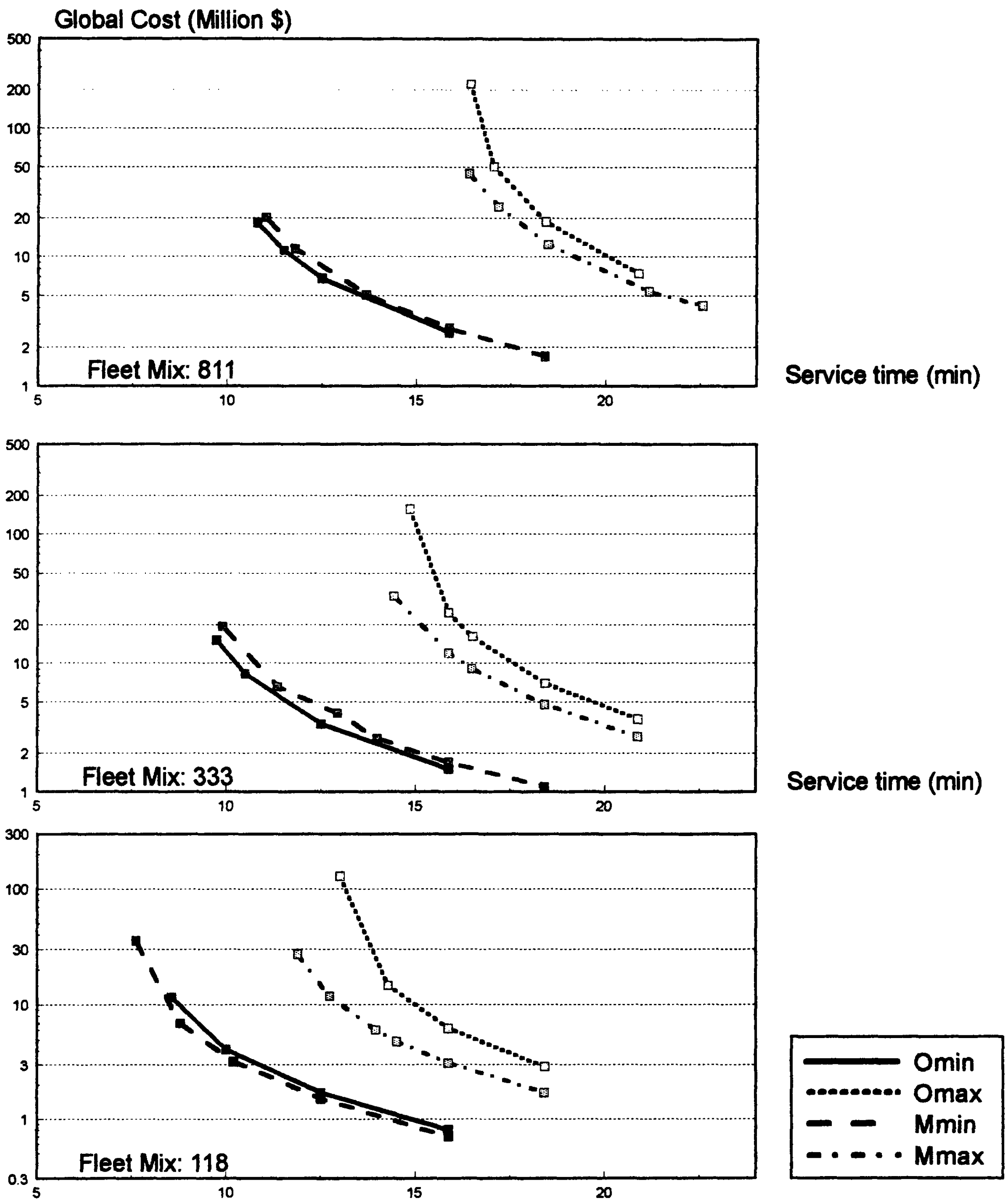
MODEL RESULTS

EFFECT OF SPEED MODE ON GLOBAL WAITING

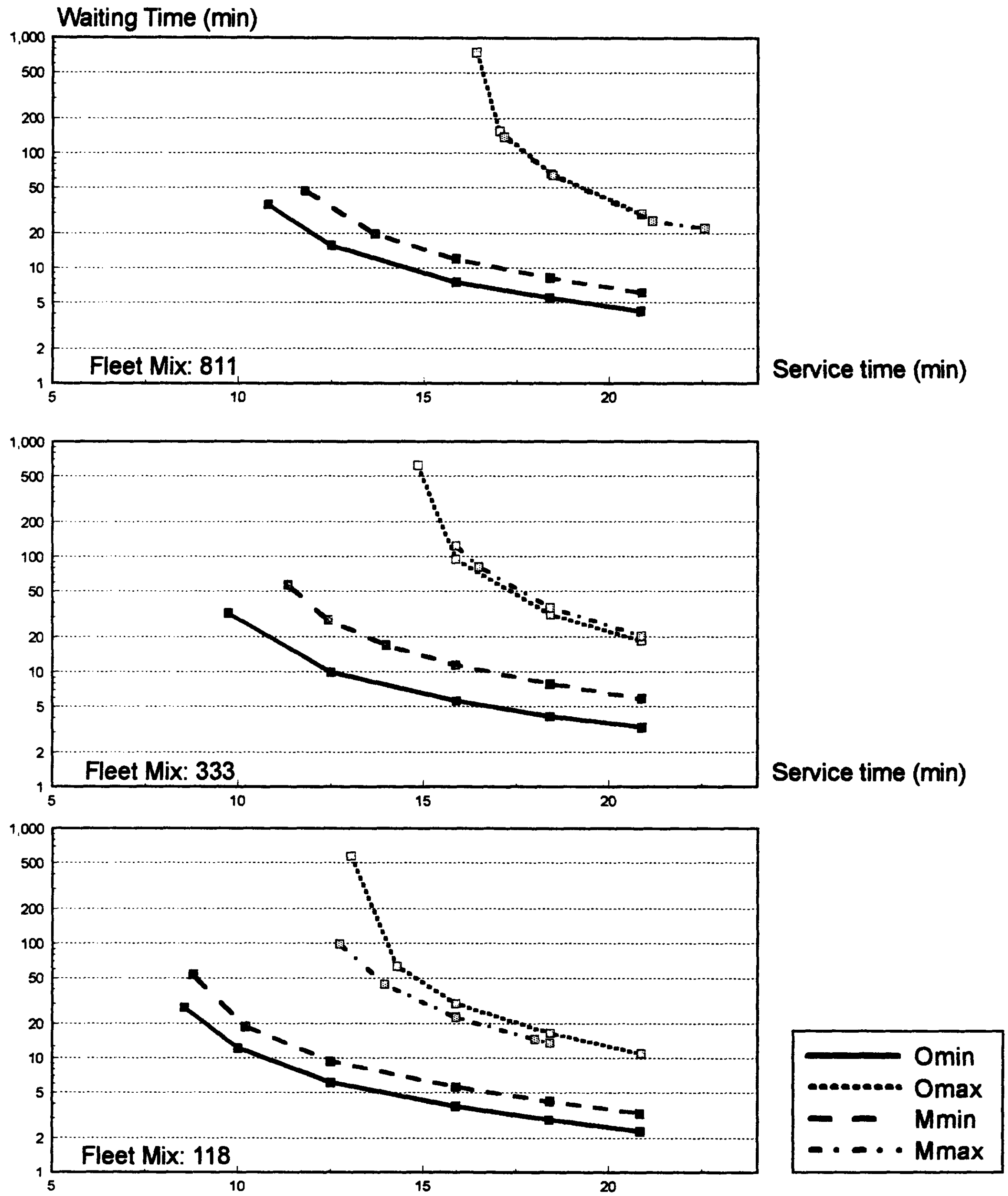
FCFS ROUTINE 2-WAY, 2.5 NM



EFFECT OF SPEED MODE ON GLOBAL COST
FCFS ROUTINE 2-WAY, 2.5 NM

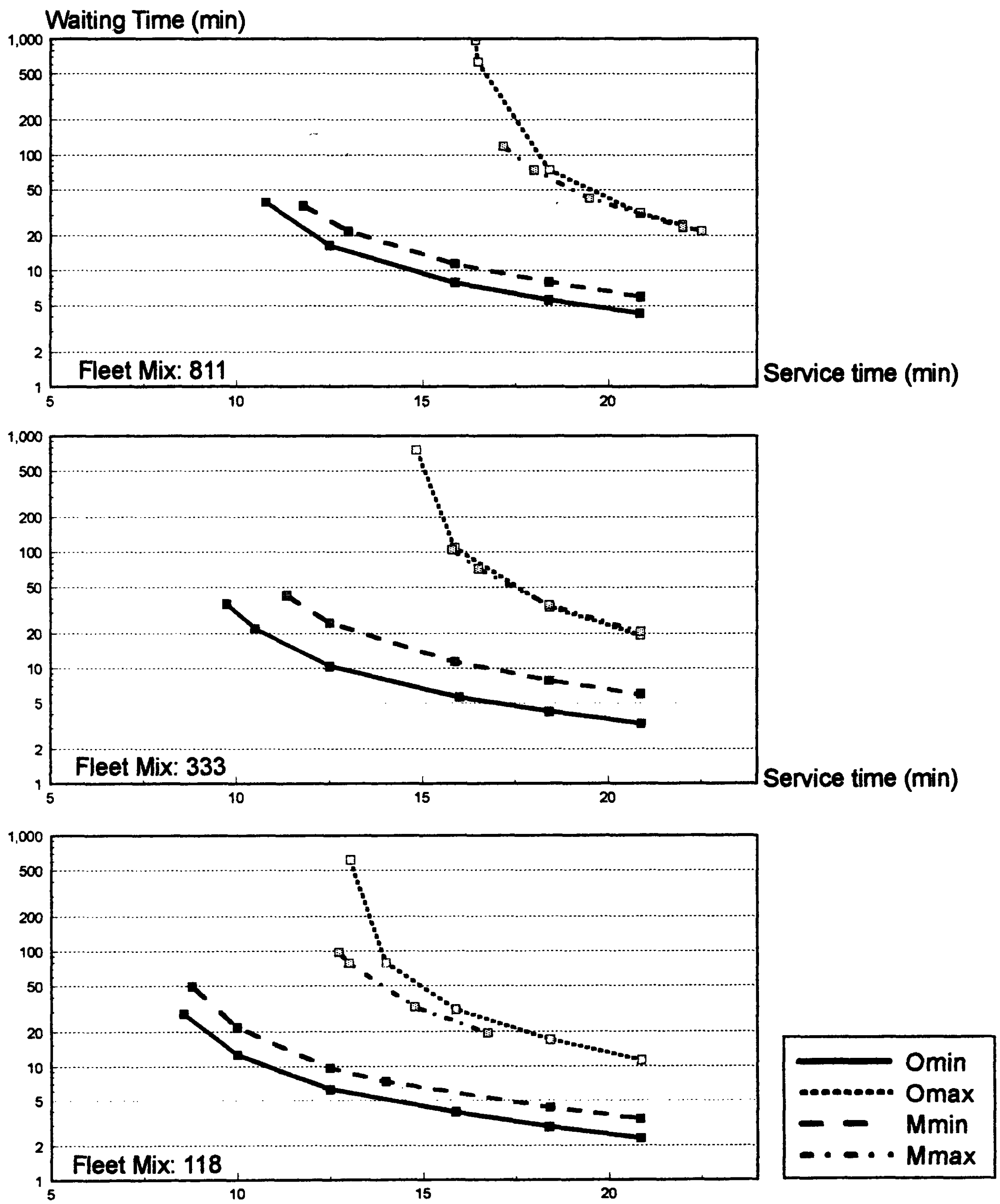


EFFECT OF SPEED MODE ON GLOBAL WAITING
FCFS ROUTINE 2-WAY, 5 NM

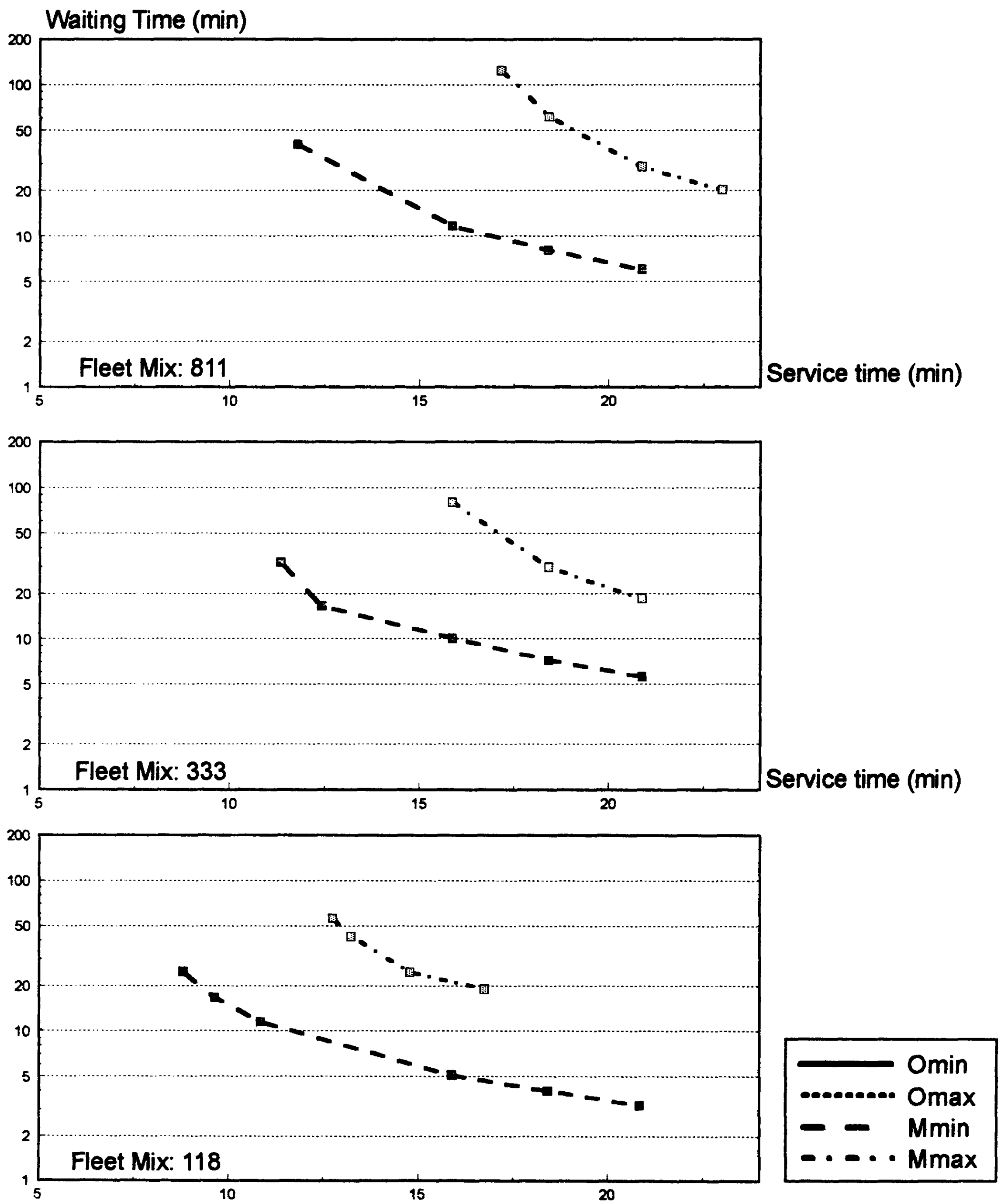


EFFECT OF SPEED MODE ON GLOBAL WAITING

COST ROUTINE 2 WAY, 5NM

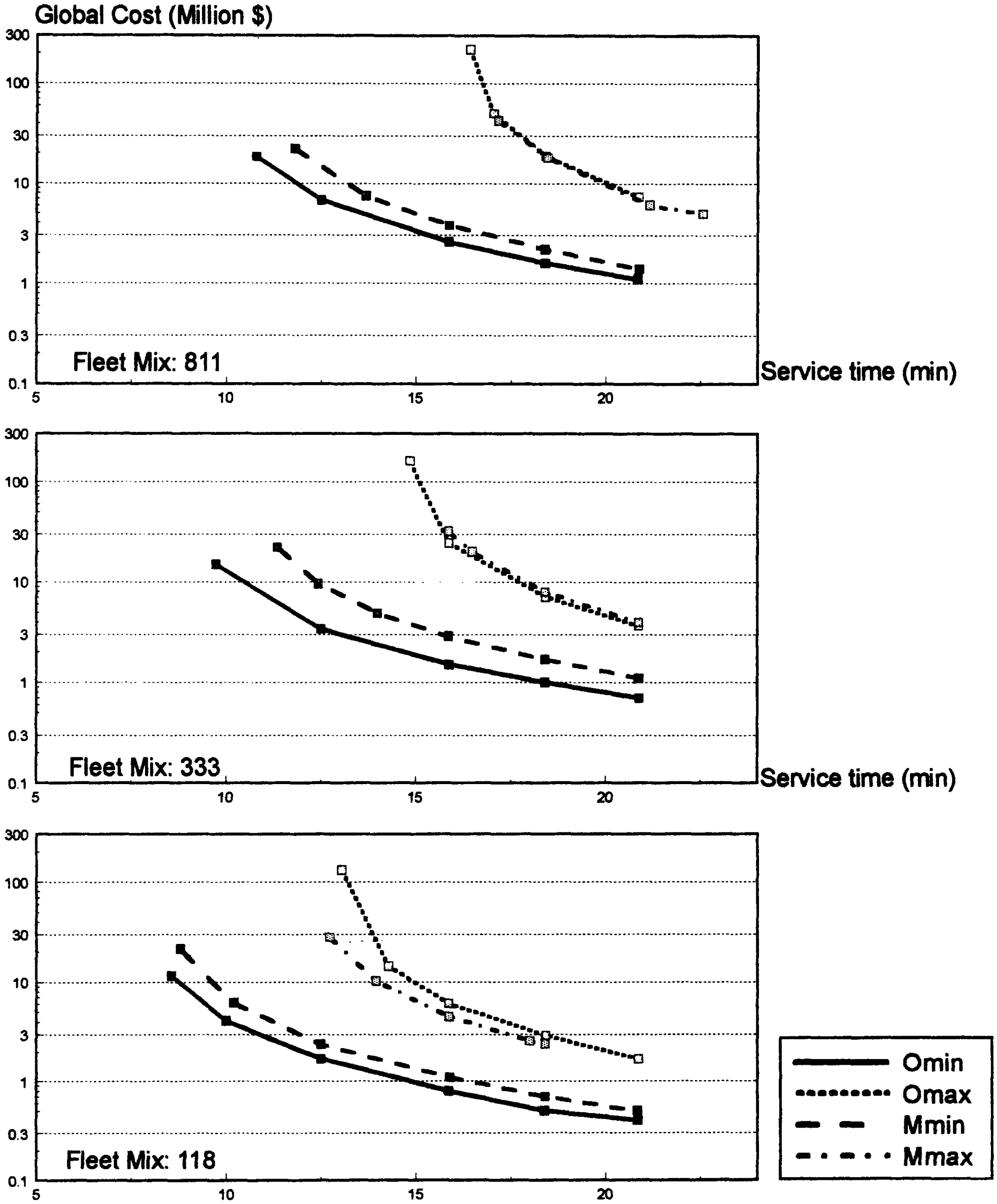


EFFECT OF SPEED MODE ON GLOBAL WAITING
FAST ROUTINE 2 WAY. 5 NM

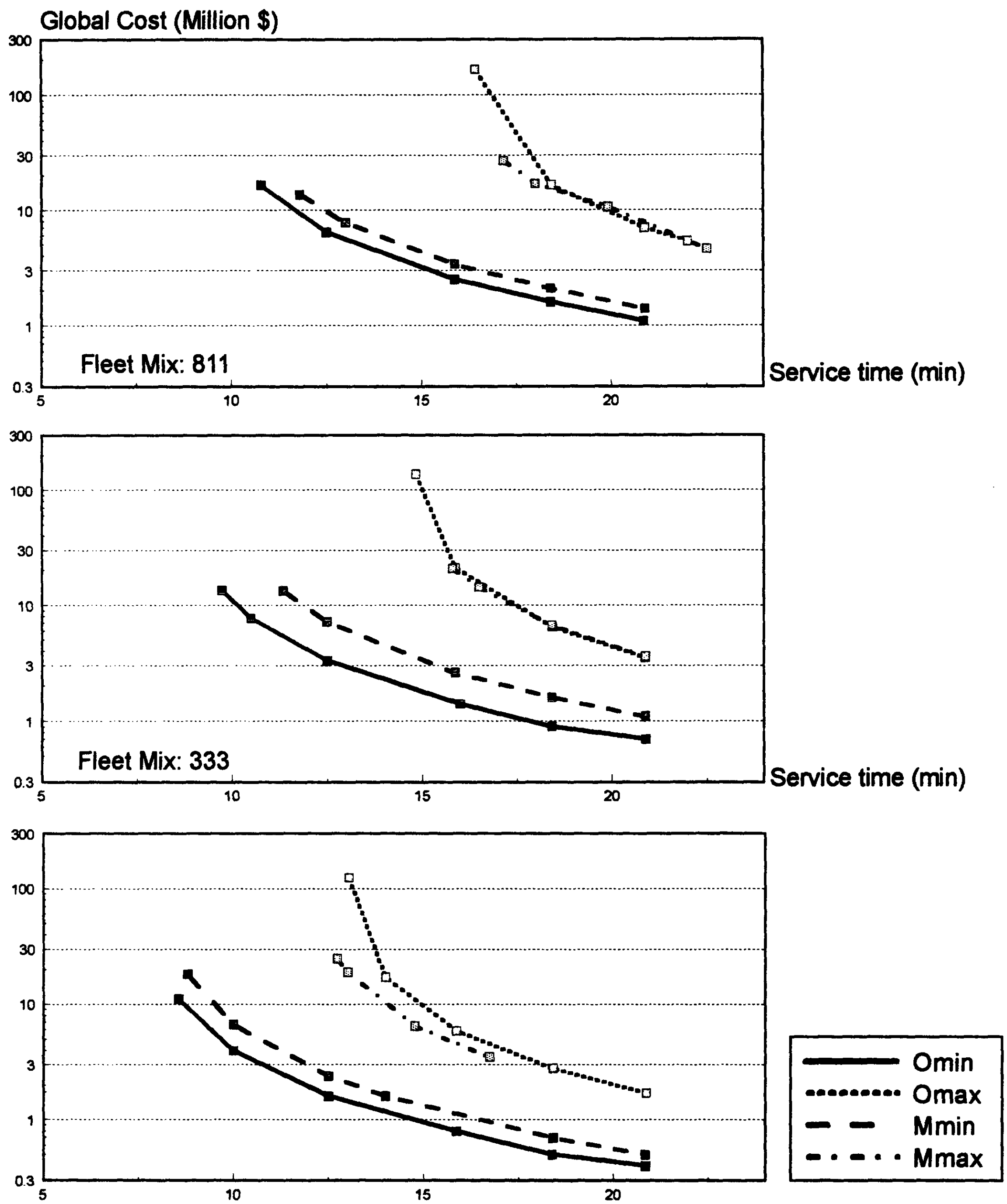


EFFECT OF SPEED MODE ON GLOBAL COST

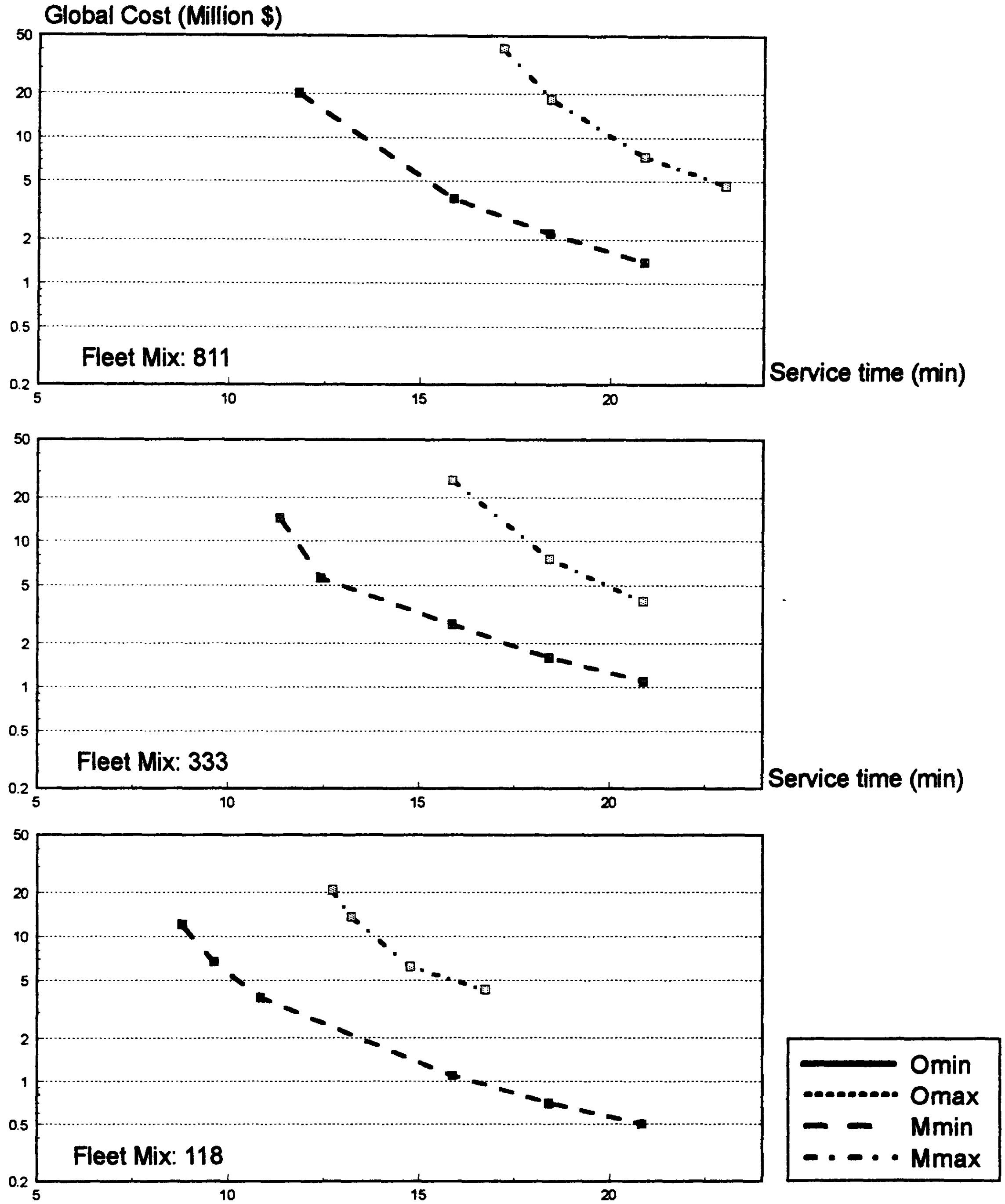
FCFS ROUTINE 2 WAY, 5 NM



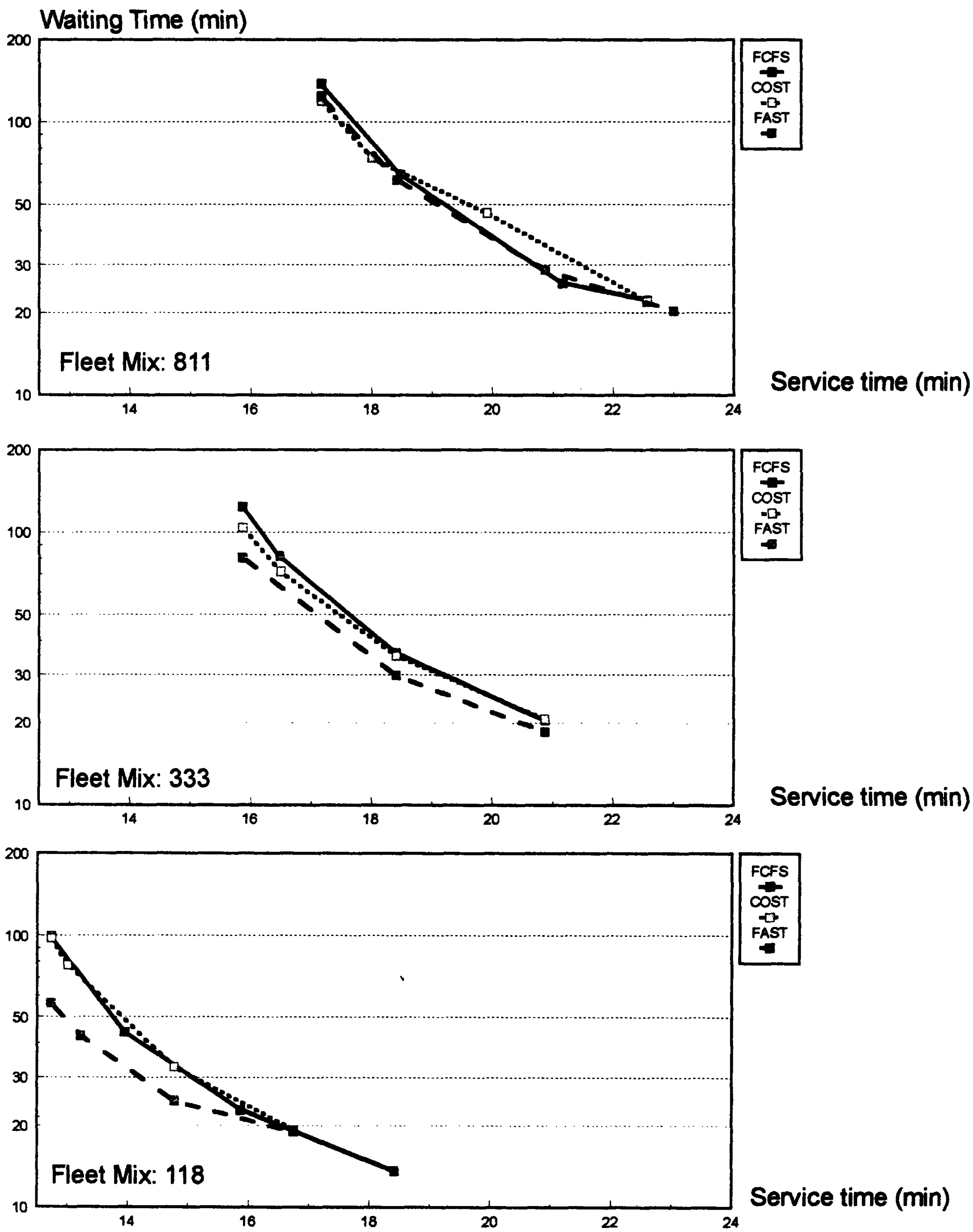
EFFECT OF SPEED MODE ON
GLOBAL COST
COST ROUTINE 2-WAY, 5 NM



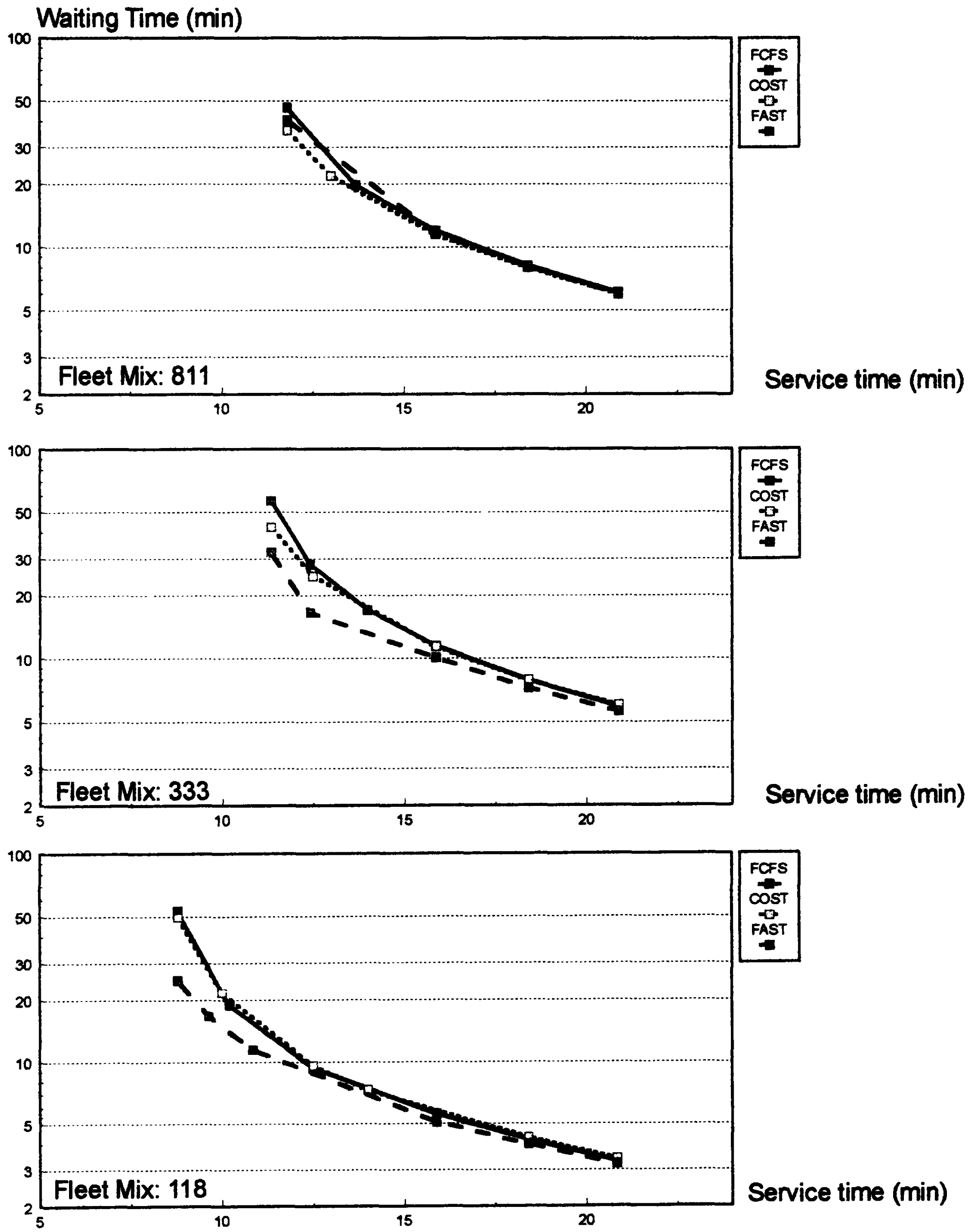
EFFECT OF SPEED MODE ON GLOBAL COST
FAST ROUTINE 2 WAY, 5 NM



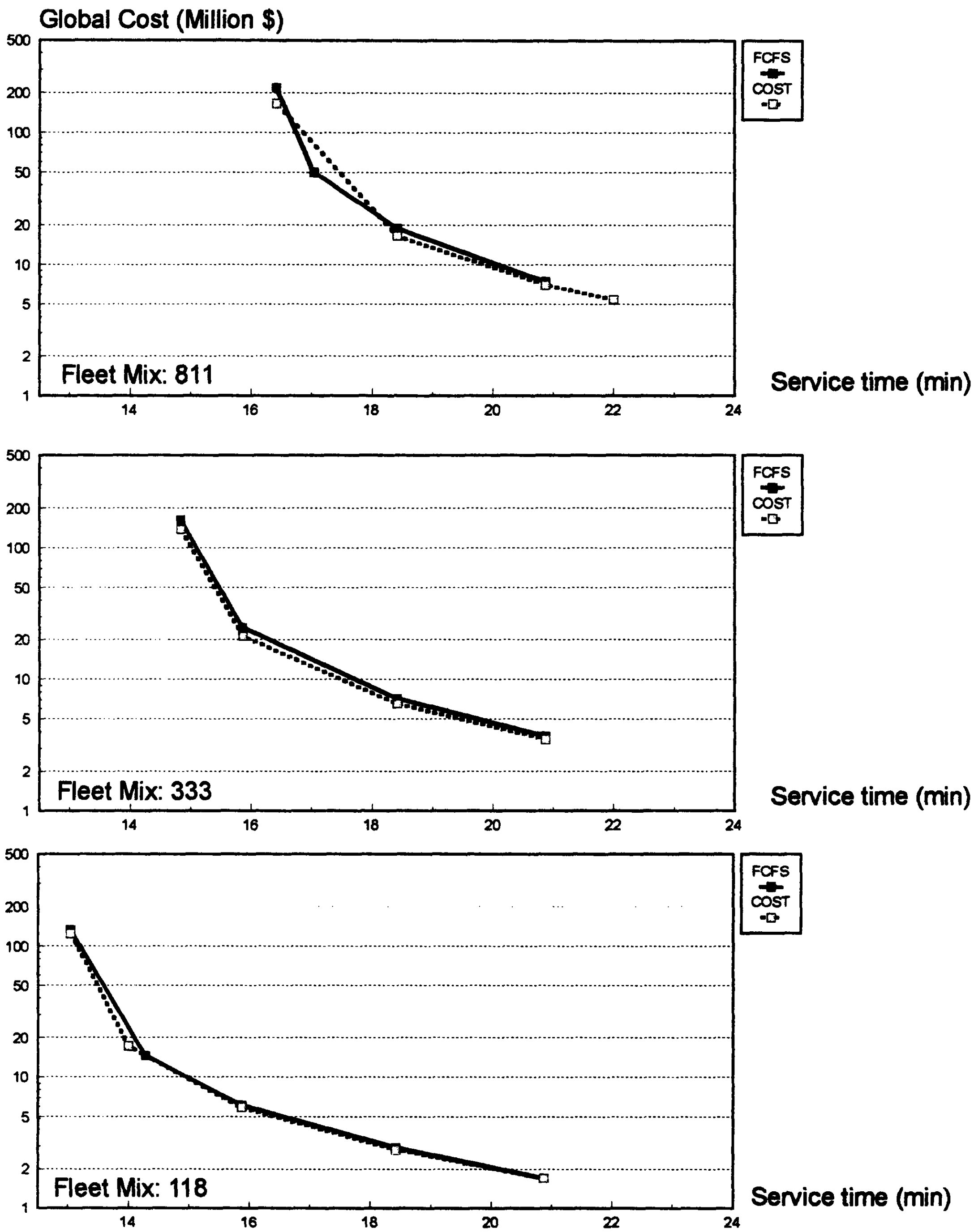
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
MULTI SPEED - MAXIMUM DISTANCE
2 WAY, 5NM



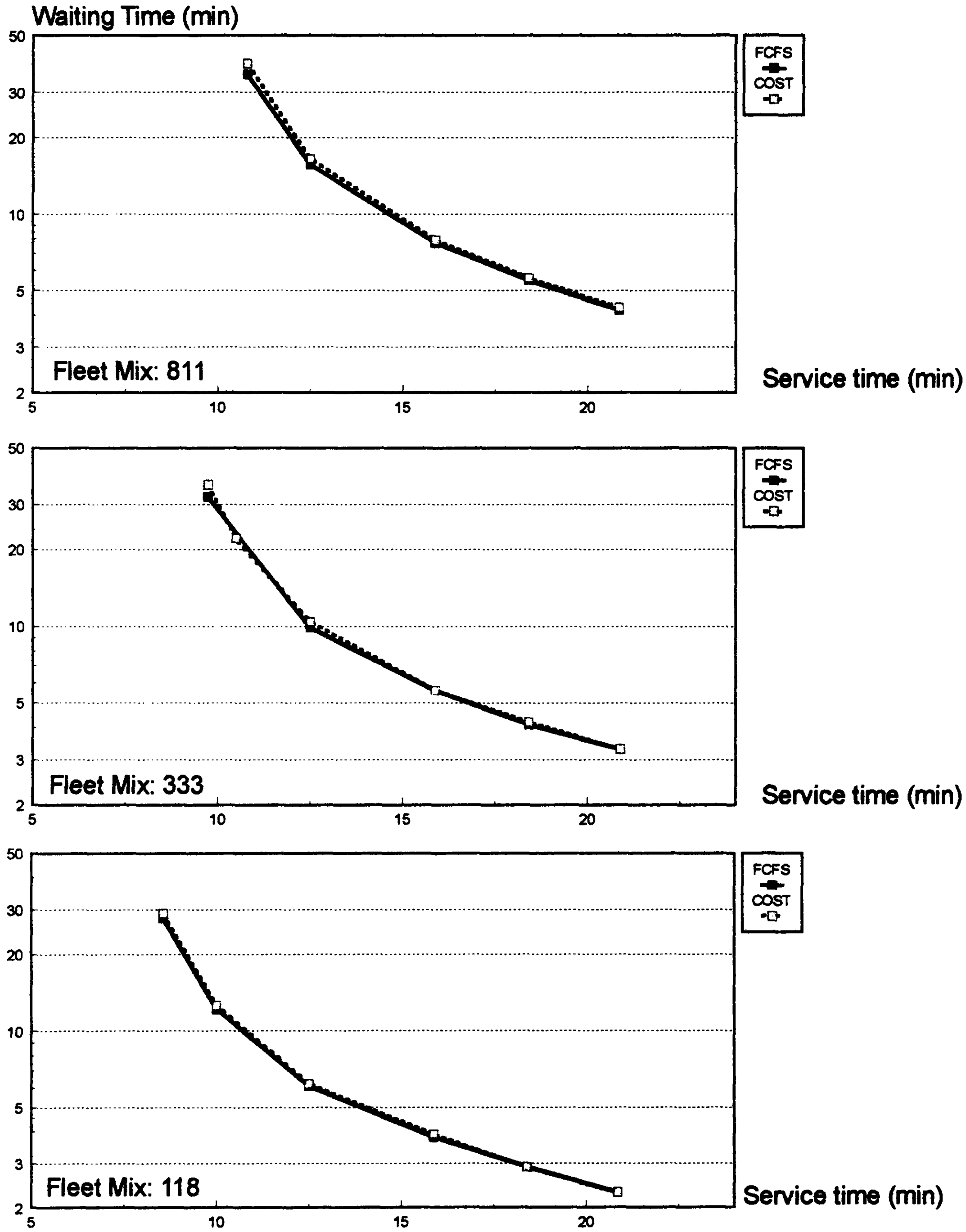
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
MULTI SPEED - MINIMUM DISTANCE
2 WAY, 5 NM



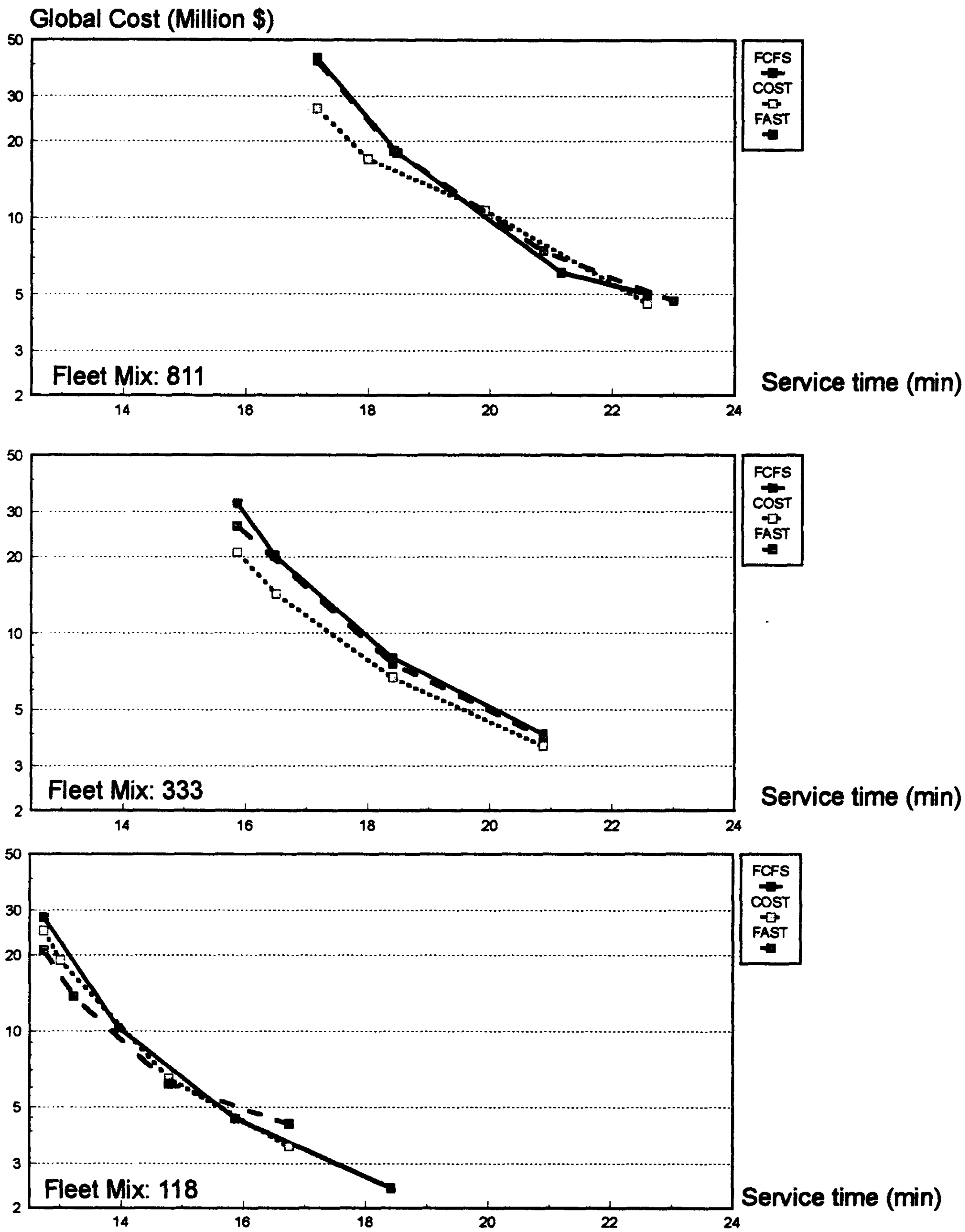
EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
ONE SPEED - MAXIMUM DISTANCE
2 WAY, 5 NM



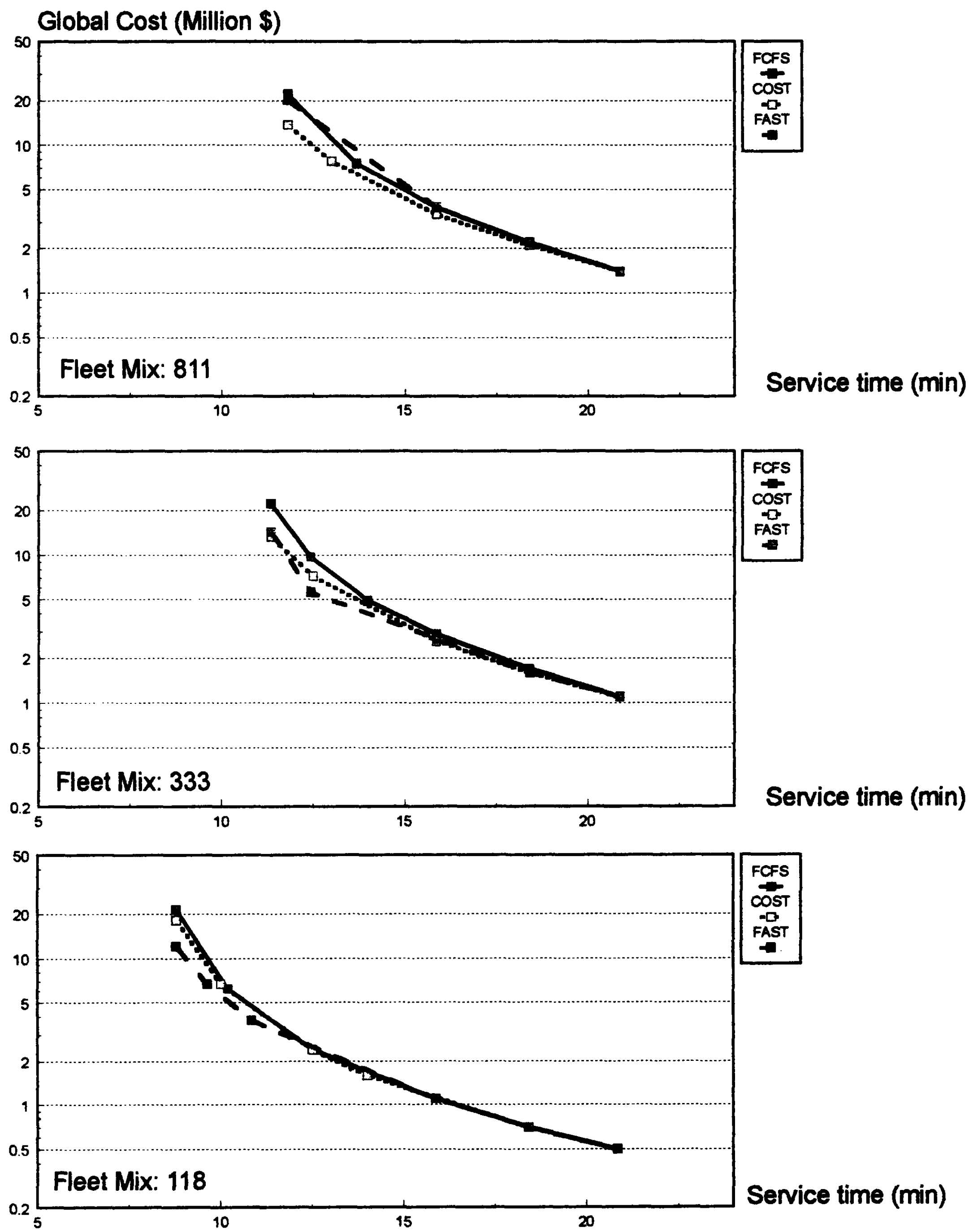
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
ONE SPEED - MINIMUM DISTANCE
2 WAY, 5 NM



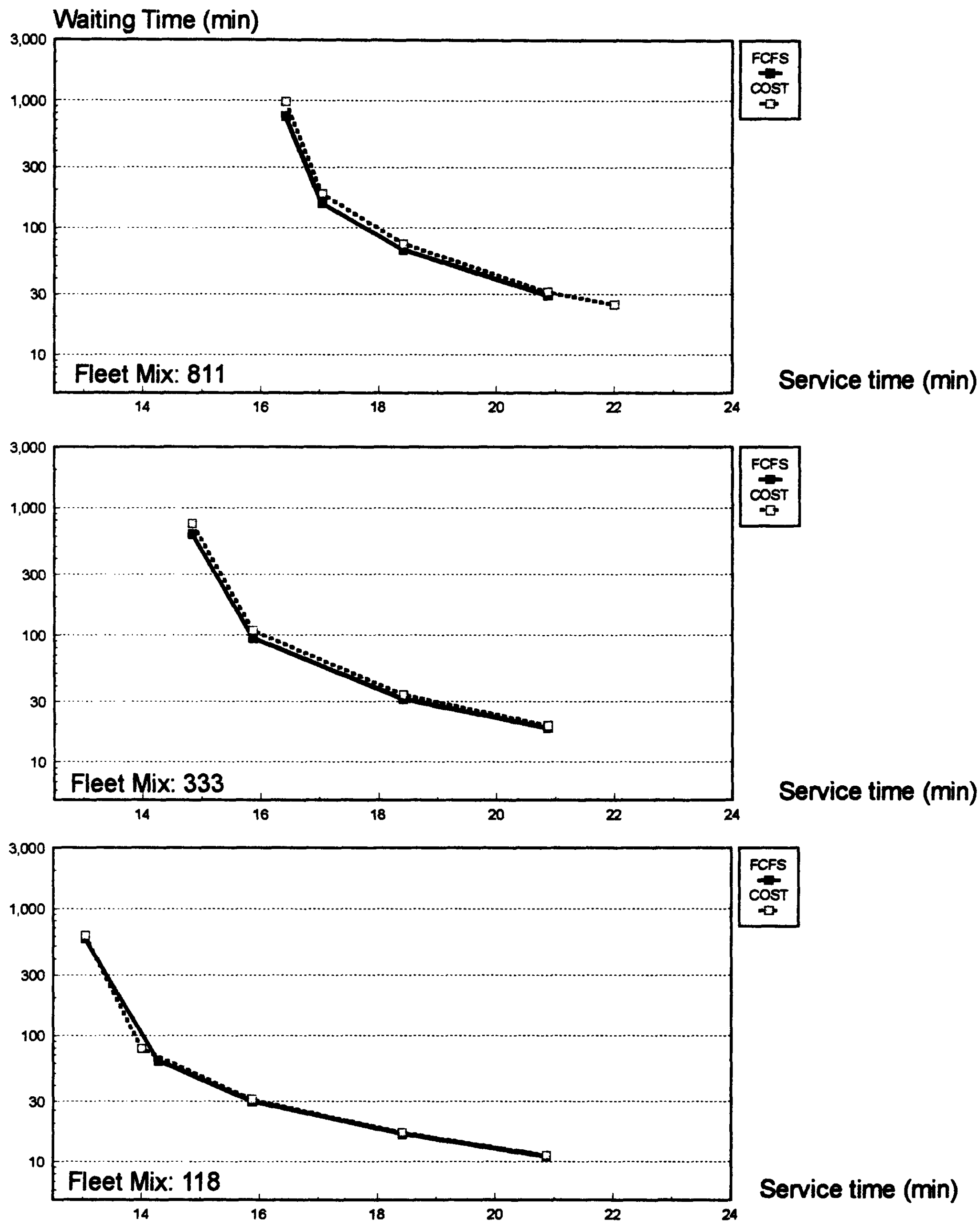
EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
MULTI SPEED - MAXIMUM DISTANCE
2 WAY, 5 NM



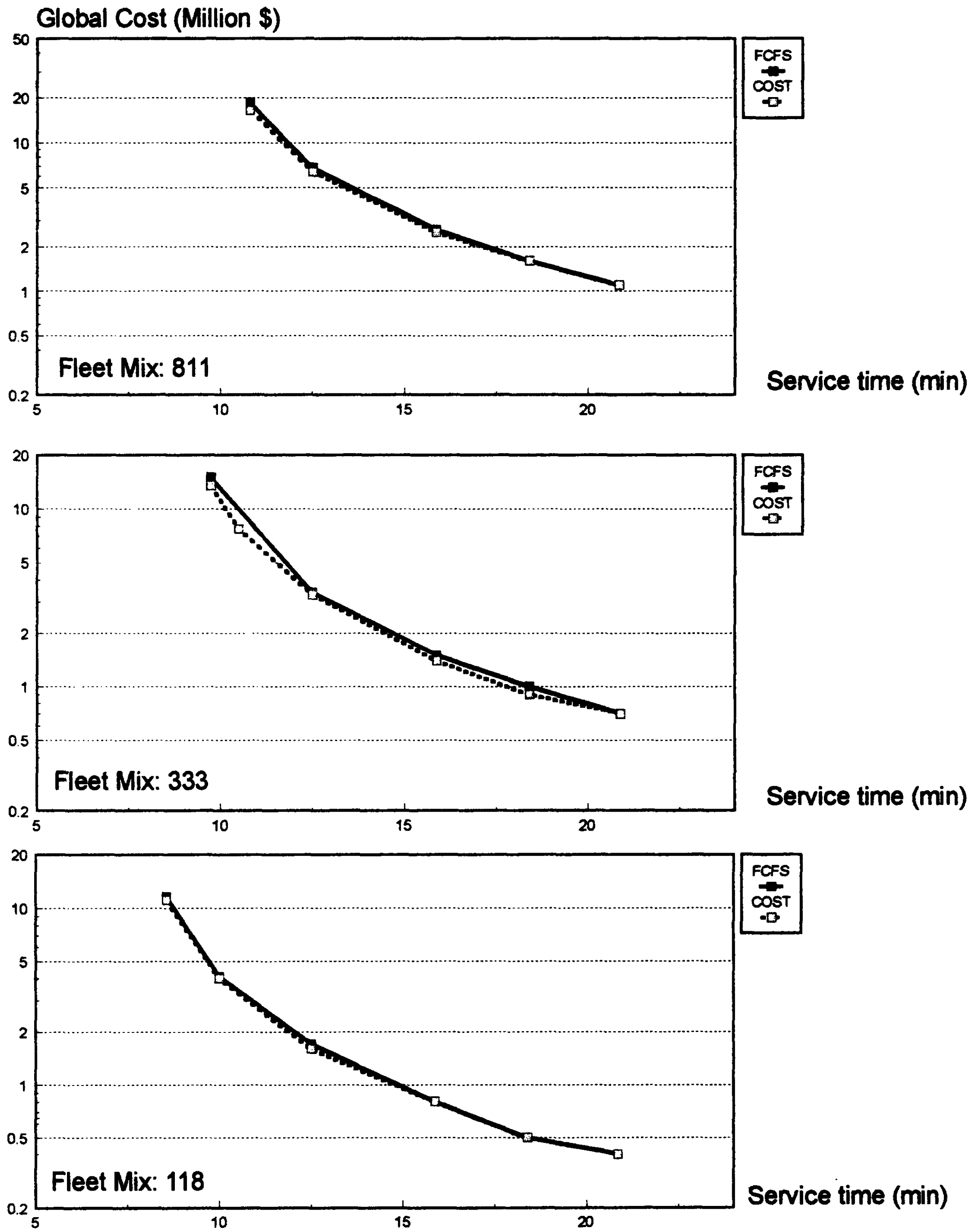
EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
MULTI SPEED - MINIMUM DISTANCE
2 WAY, 5 NM



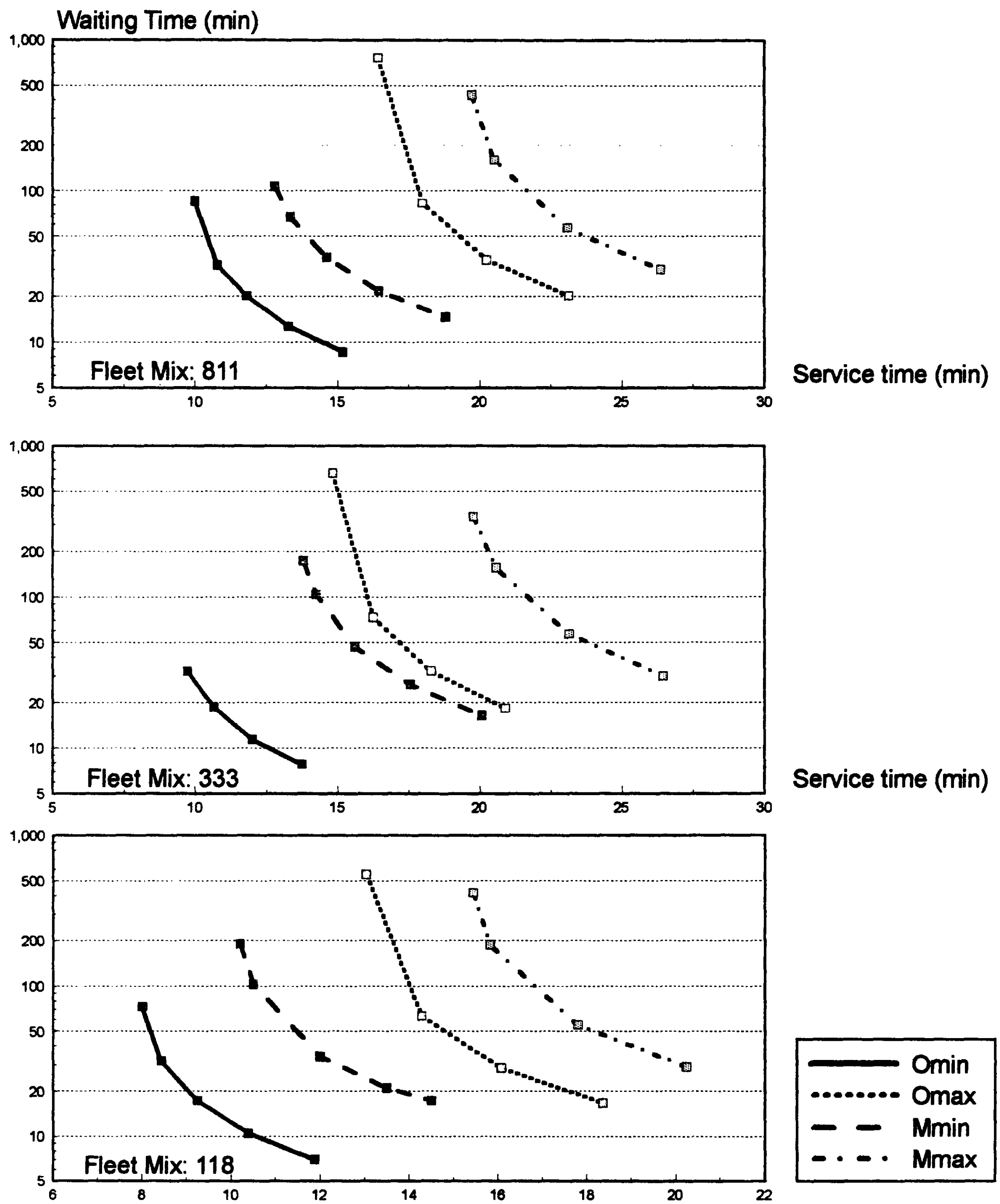
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
ONE SPEED - MAXIMUM DISTANCE
2 WAY, 5 NM



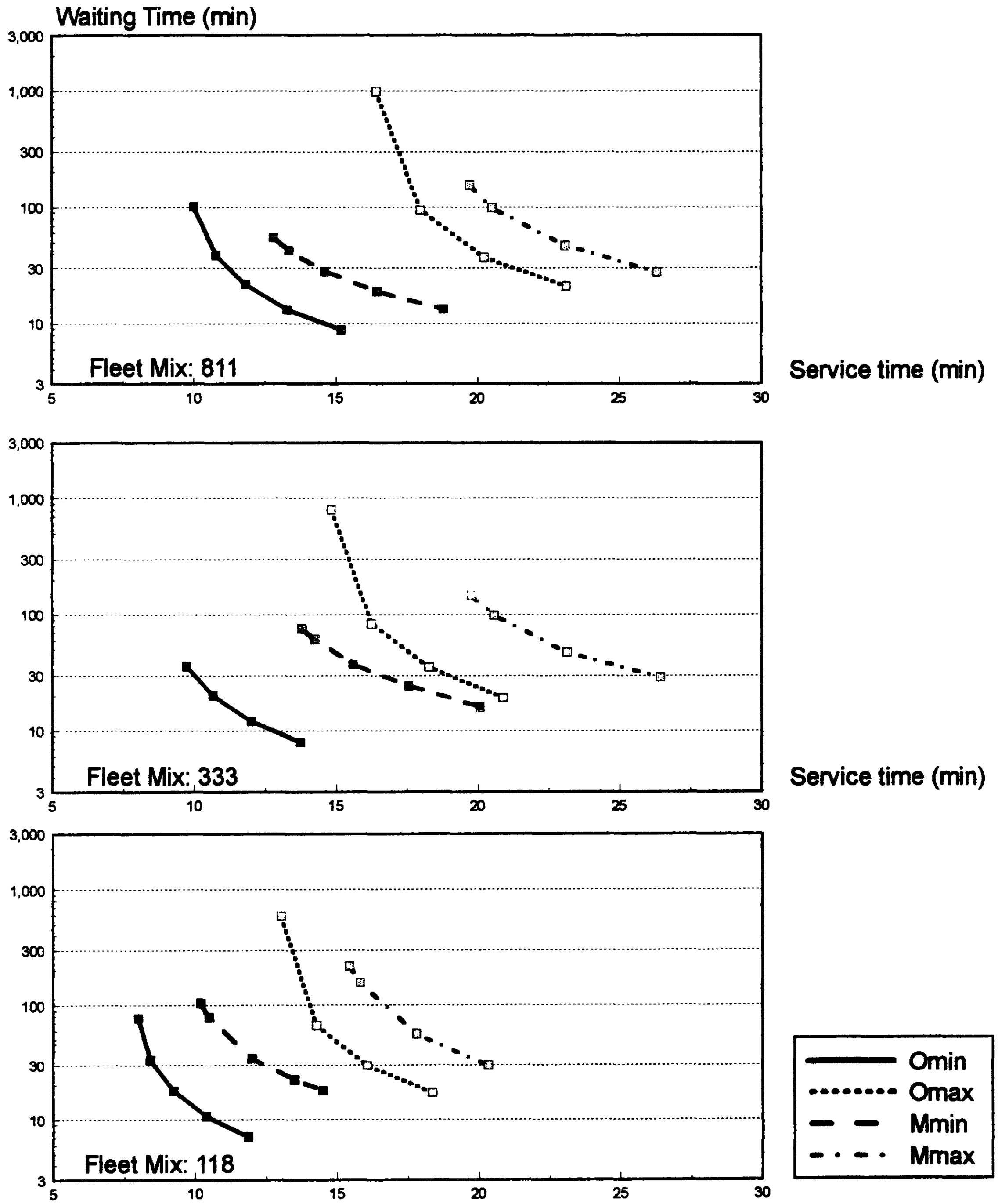
EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
ONE SPEED - MINIMUM DISTANCE
2 WAY, 5 NM



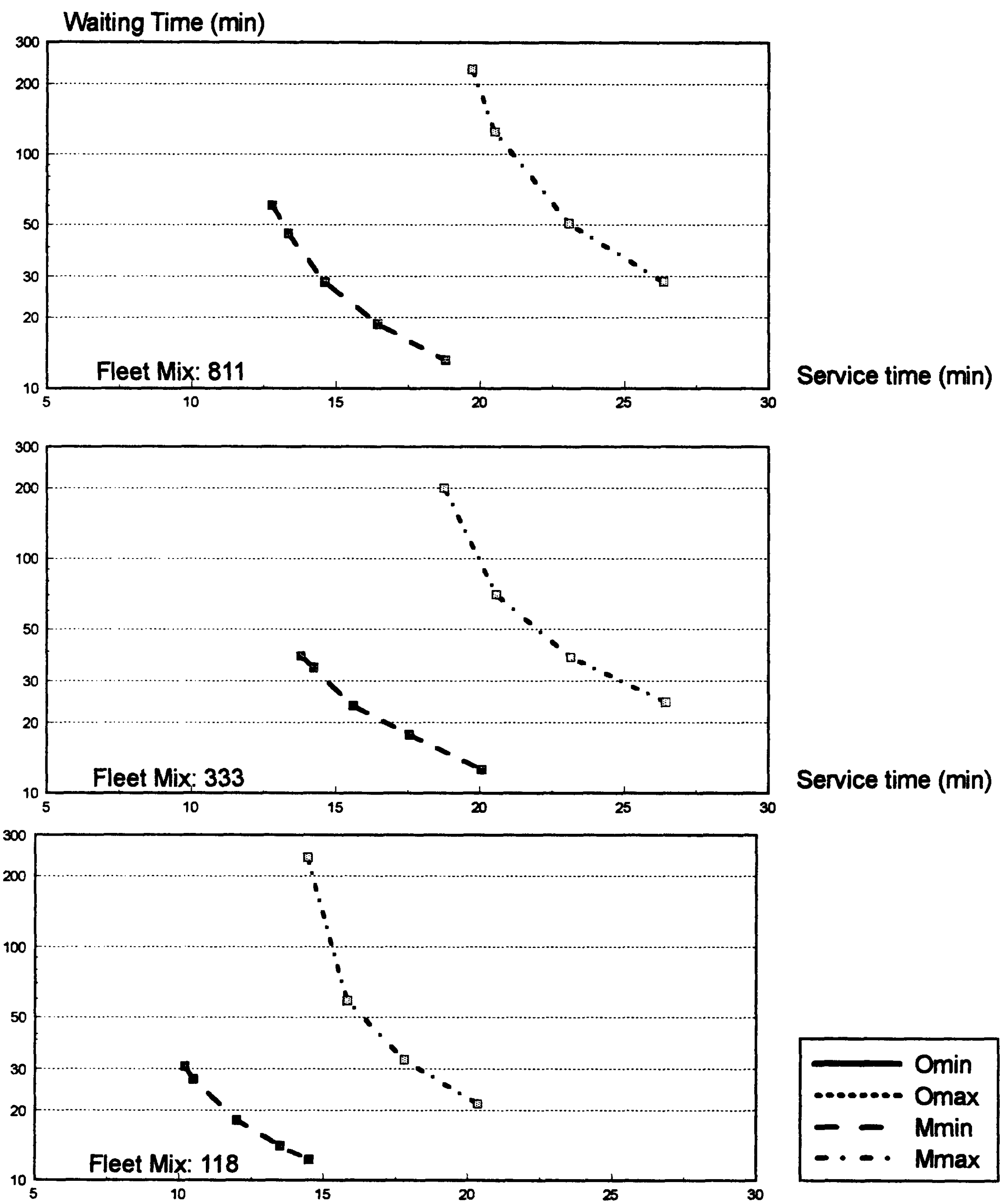
EFFECT OF SPEED MODE ON GLOBAL WAITING
FCFS ROUTINE 2-WAY, 10 NM



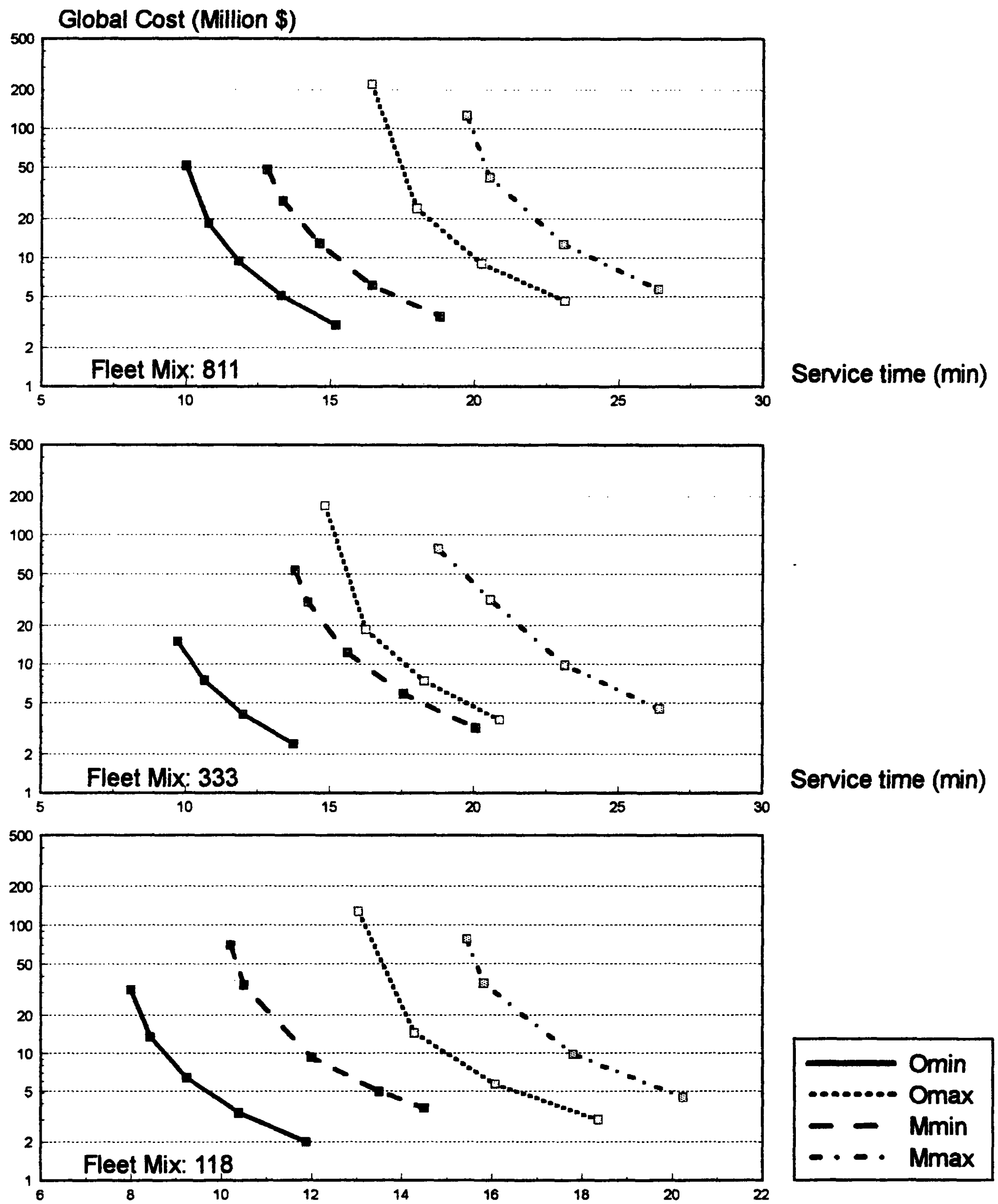
EFFECT OF SPEED MODE ON
GLOBAL WAITING
COST ROUTINE 2-WAY, 10 NM



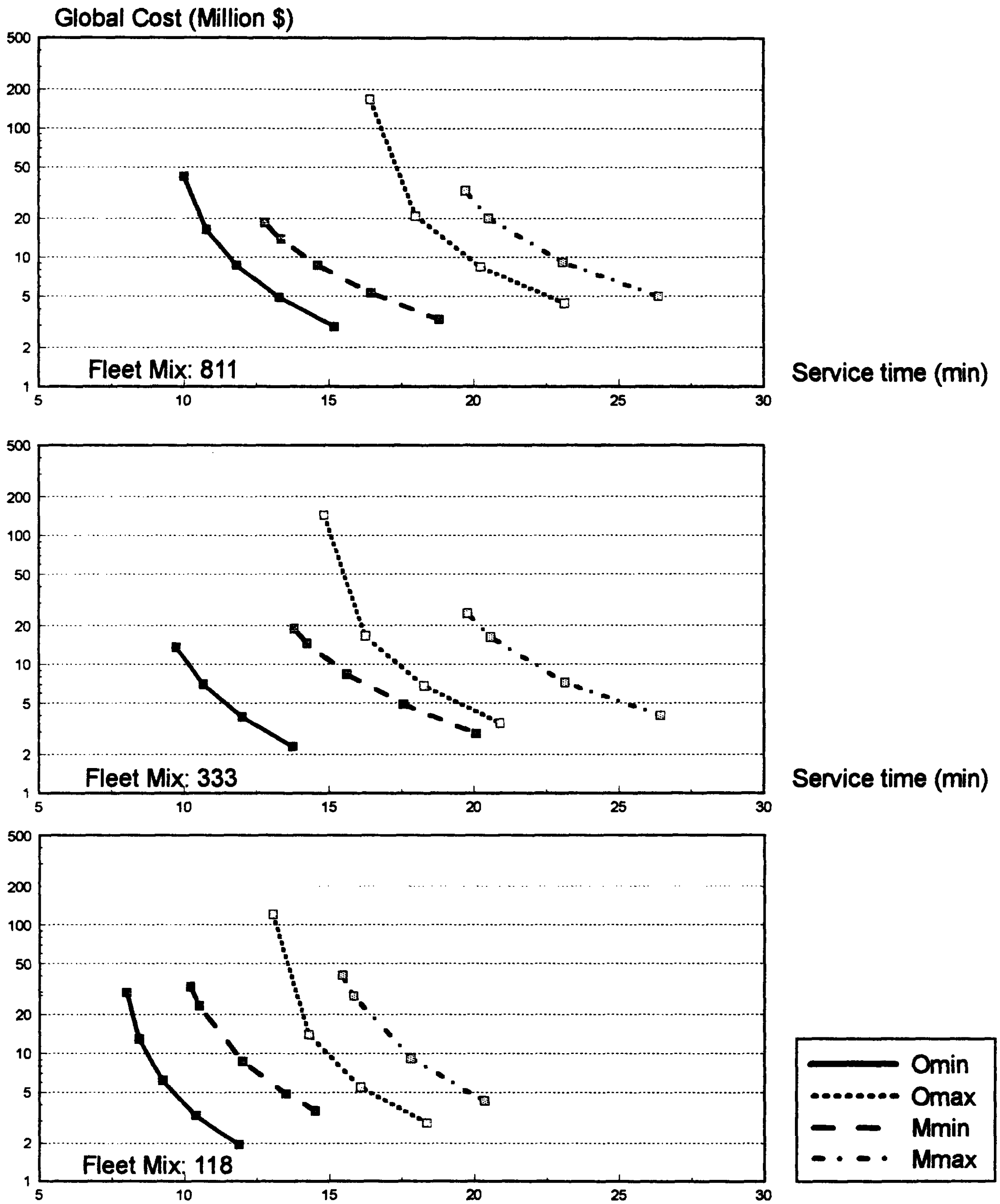
EFFECT OF SPEED MODE ON GLOBAL WAITING
FAST ROUTINE 2-WAY, 10 NM



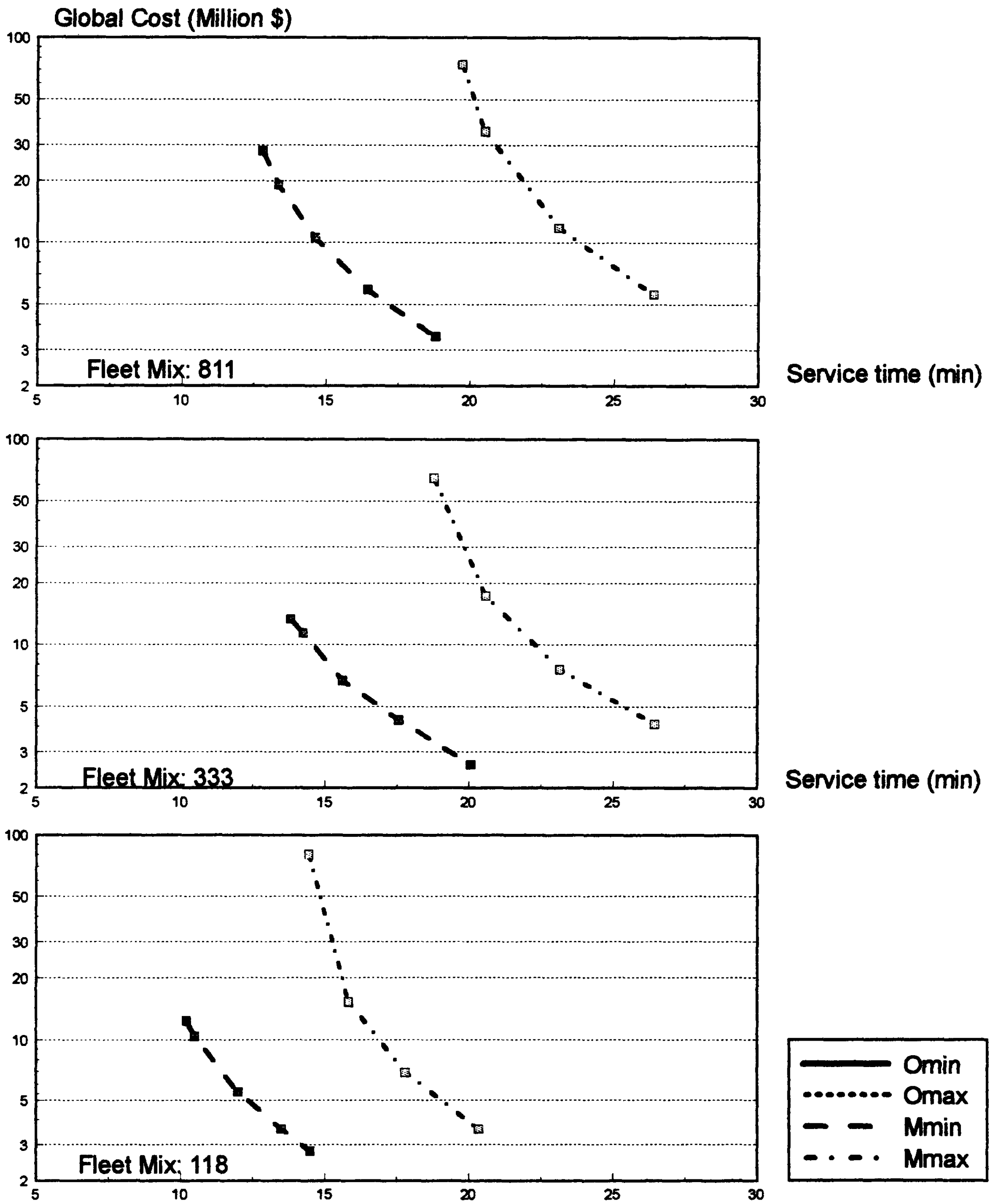
EFFECT OF SPEED MODE ON GLOBAL COST
FCFS ROUTINE 2-WAY, 10 NM2



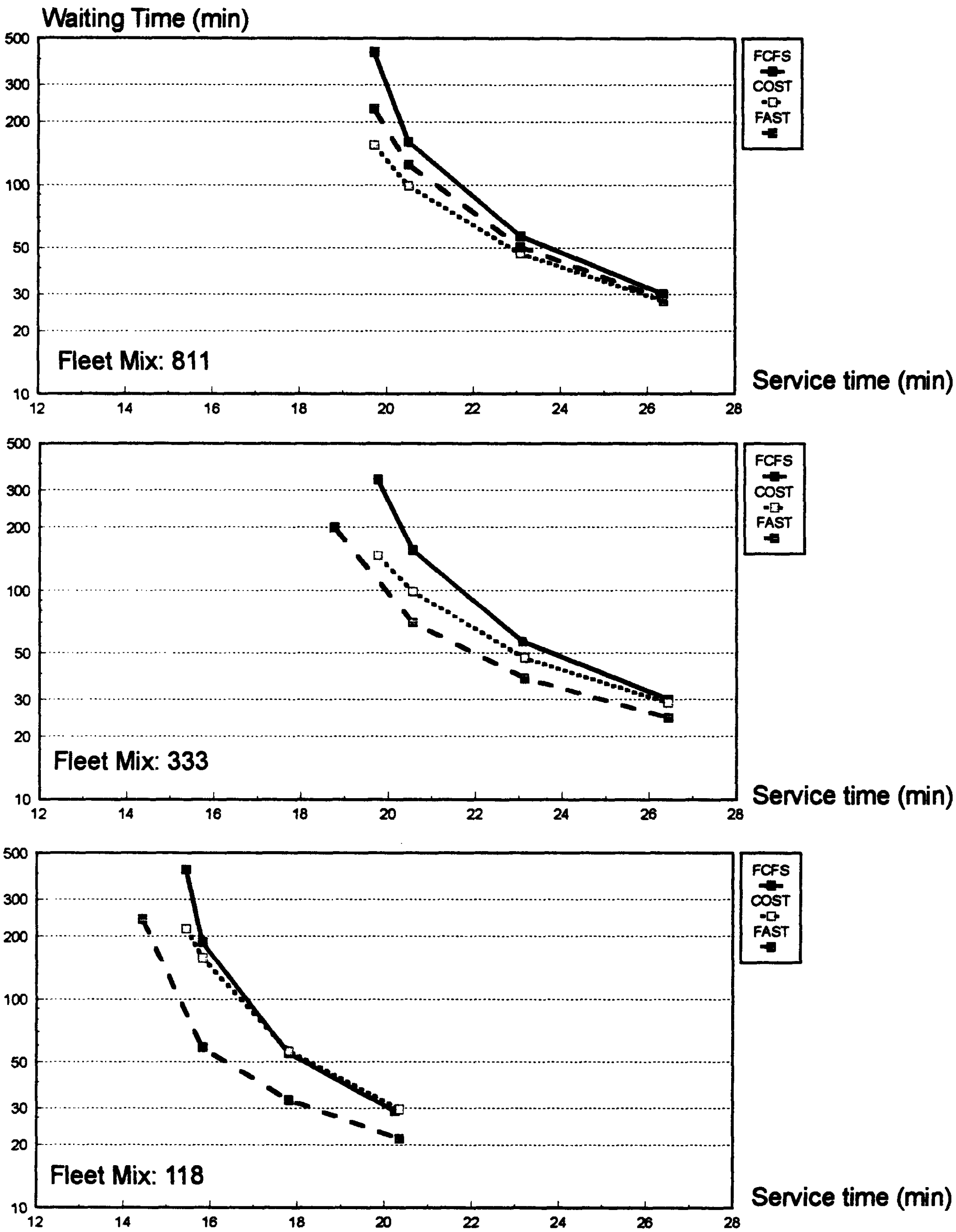
EFFECT OF SPEED MODE ON GLOBAL COST
COST ROUTINE 2-WAY, 10 NM



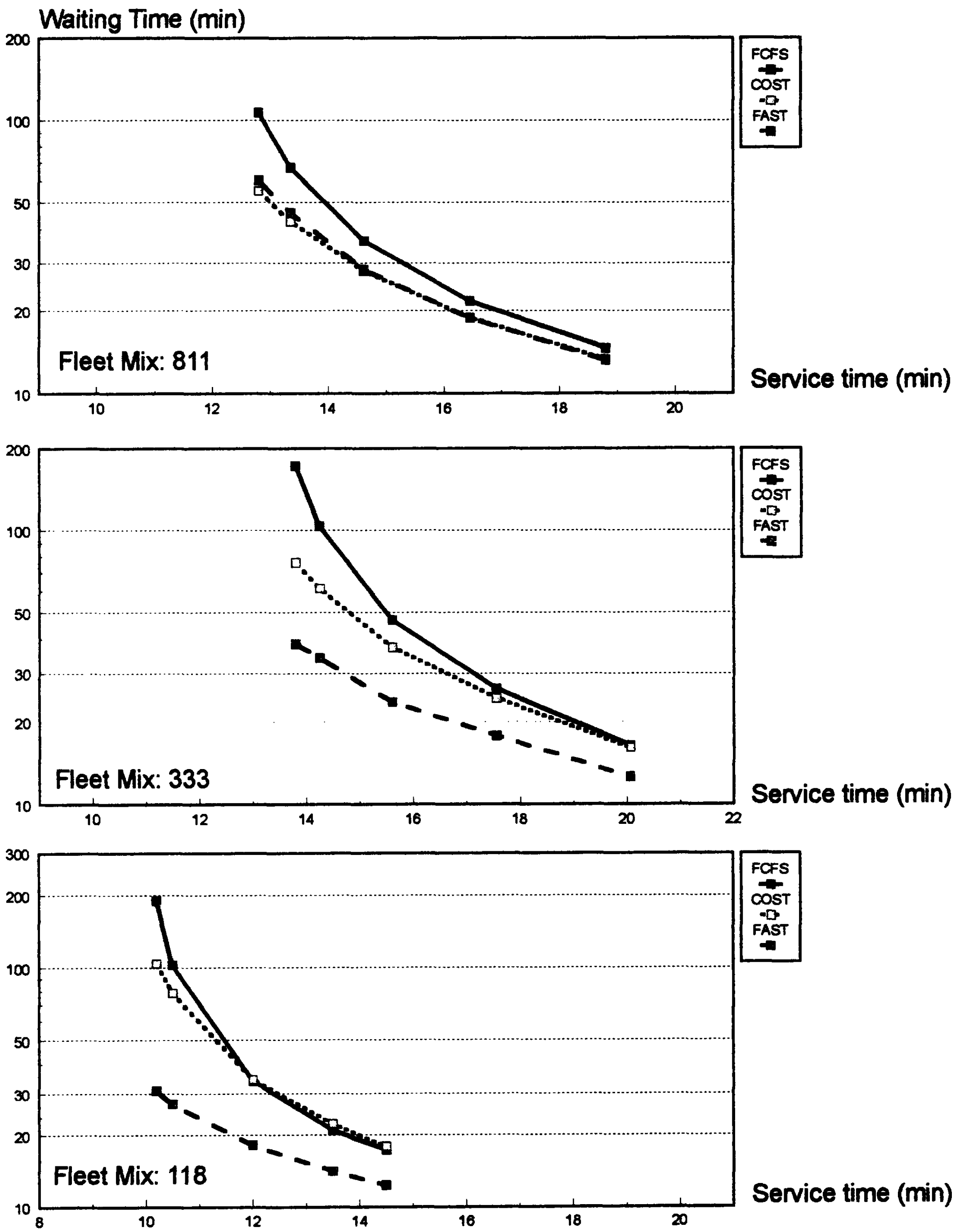
EFFECT OF SPEED MODE ON GLOBAL COST
FAST ROUTINE
2-WAY, 10 NM



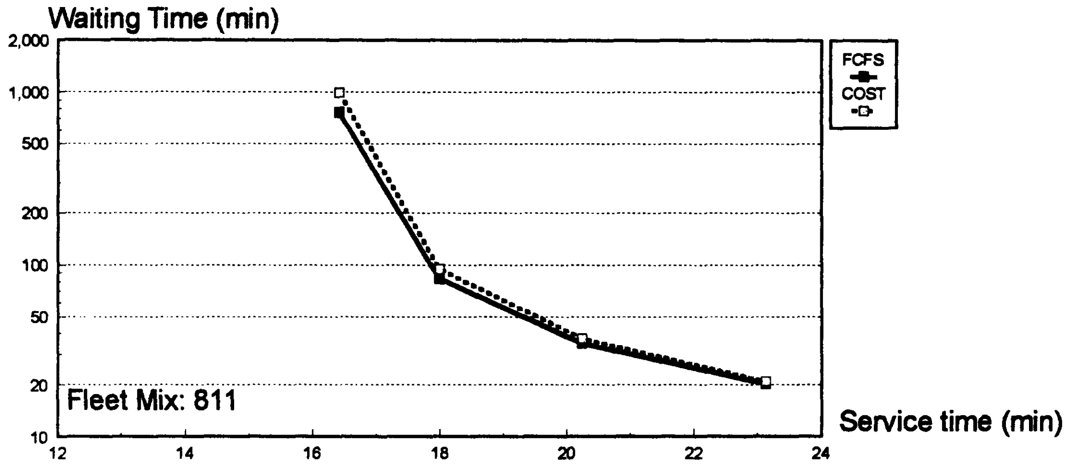
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
MULTI SPEED - MAXIMUM DISTANCE
2 WAY, 10 NM

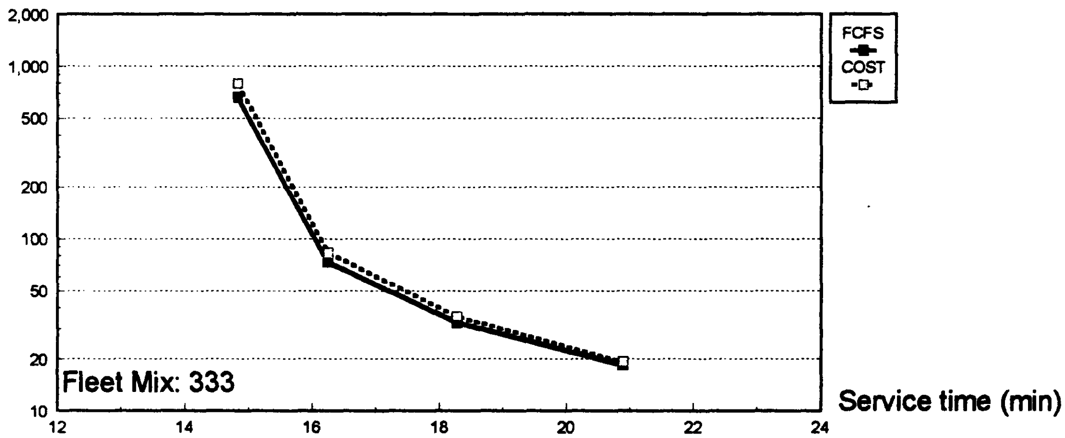


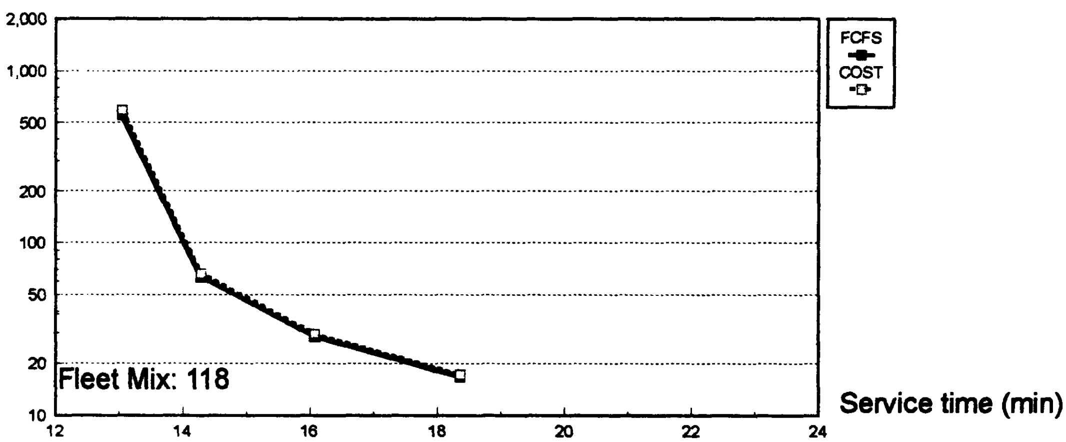
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
MULTI SPEED - MINIMUM DISTANCE
2 WAY, 10 NM



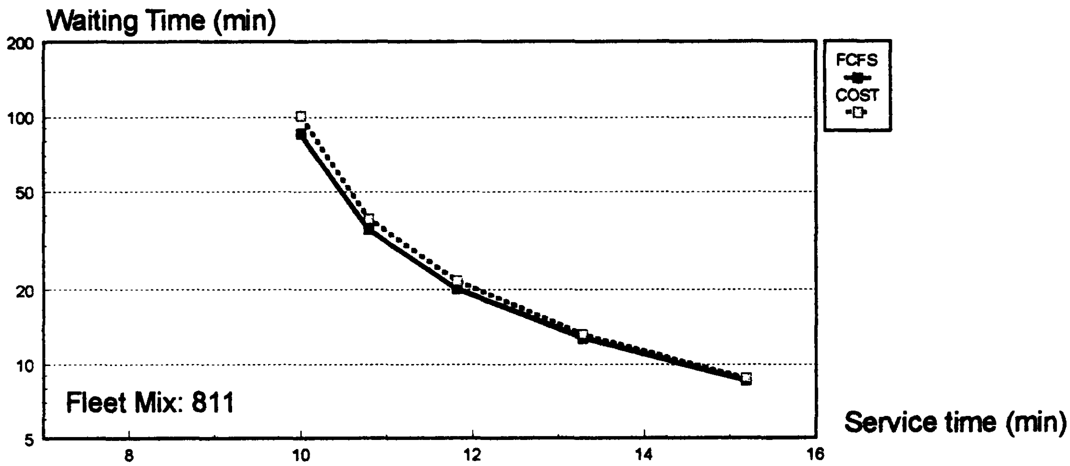
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
ONE SPEED - MAXIMUM DISTANCE
2 WAY, 10 NM

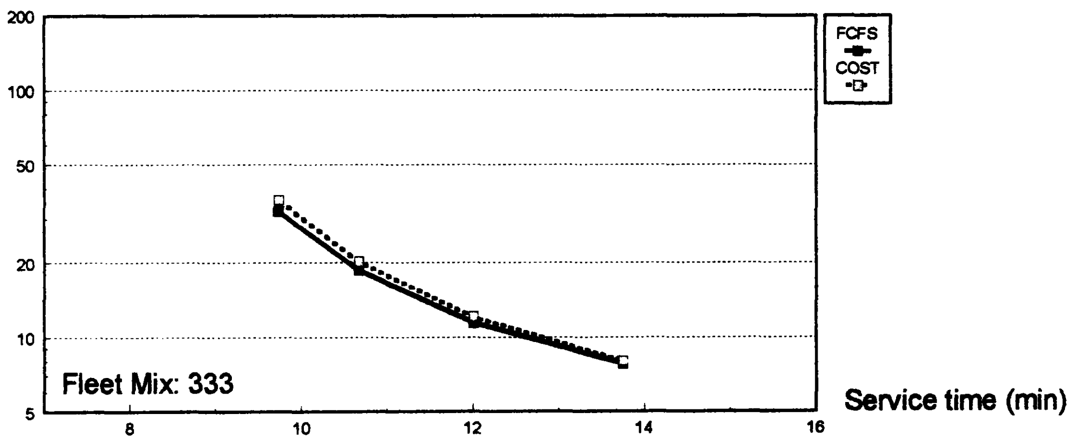


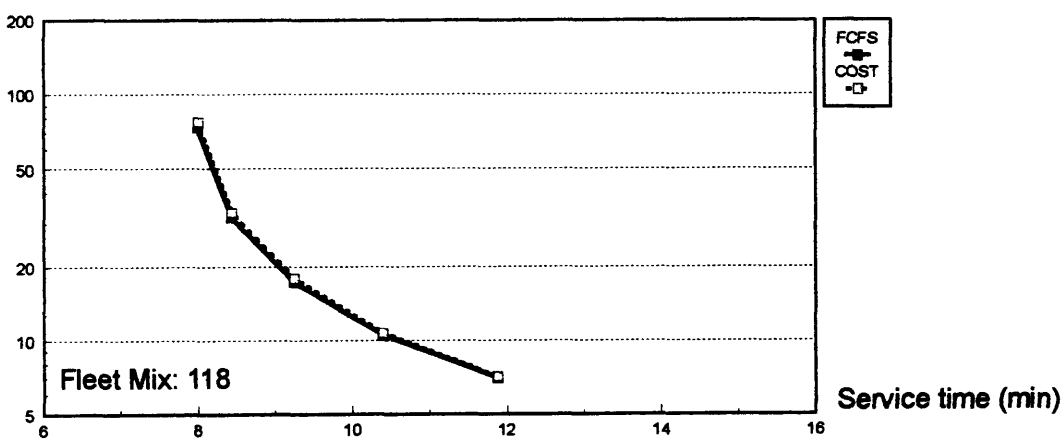




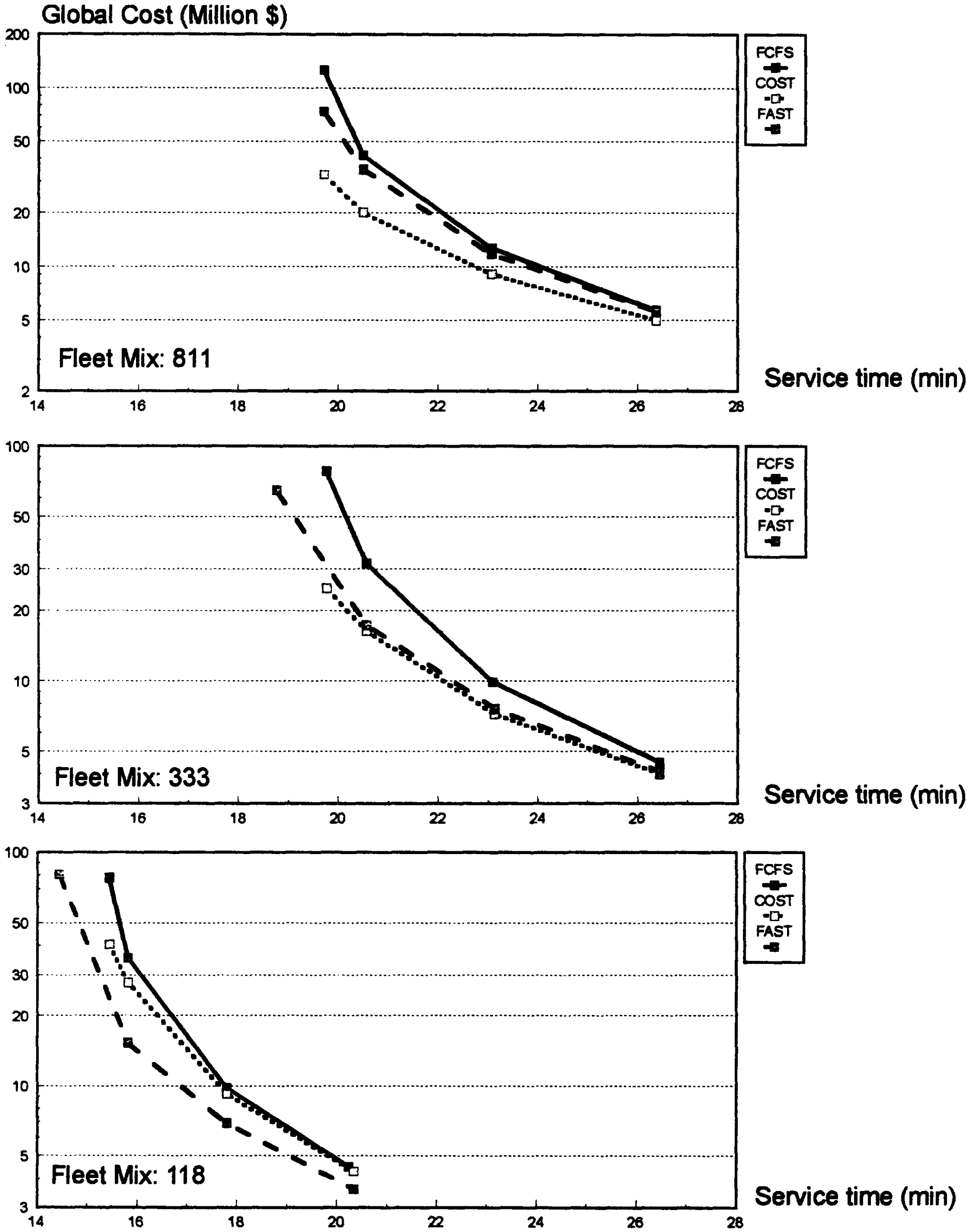
EFFECT OF QUEUEING ROUTINE IN GLOBAL WAITING
ONE SPEED - MINIMUM DISTANCE
2 WAY, 10 NM



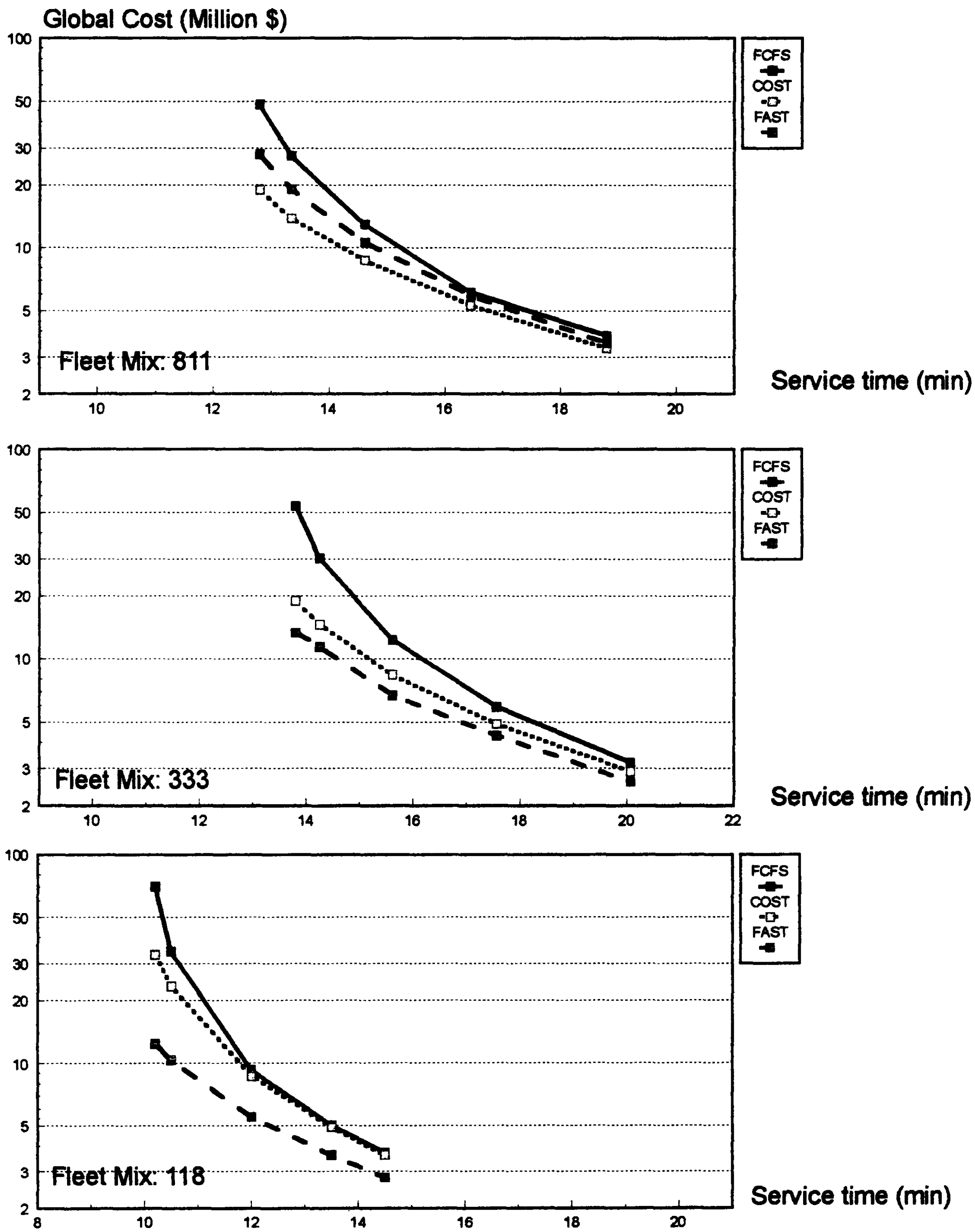




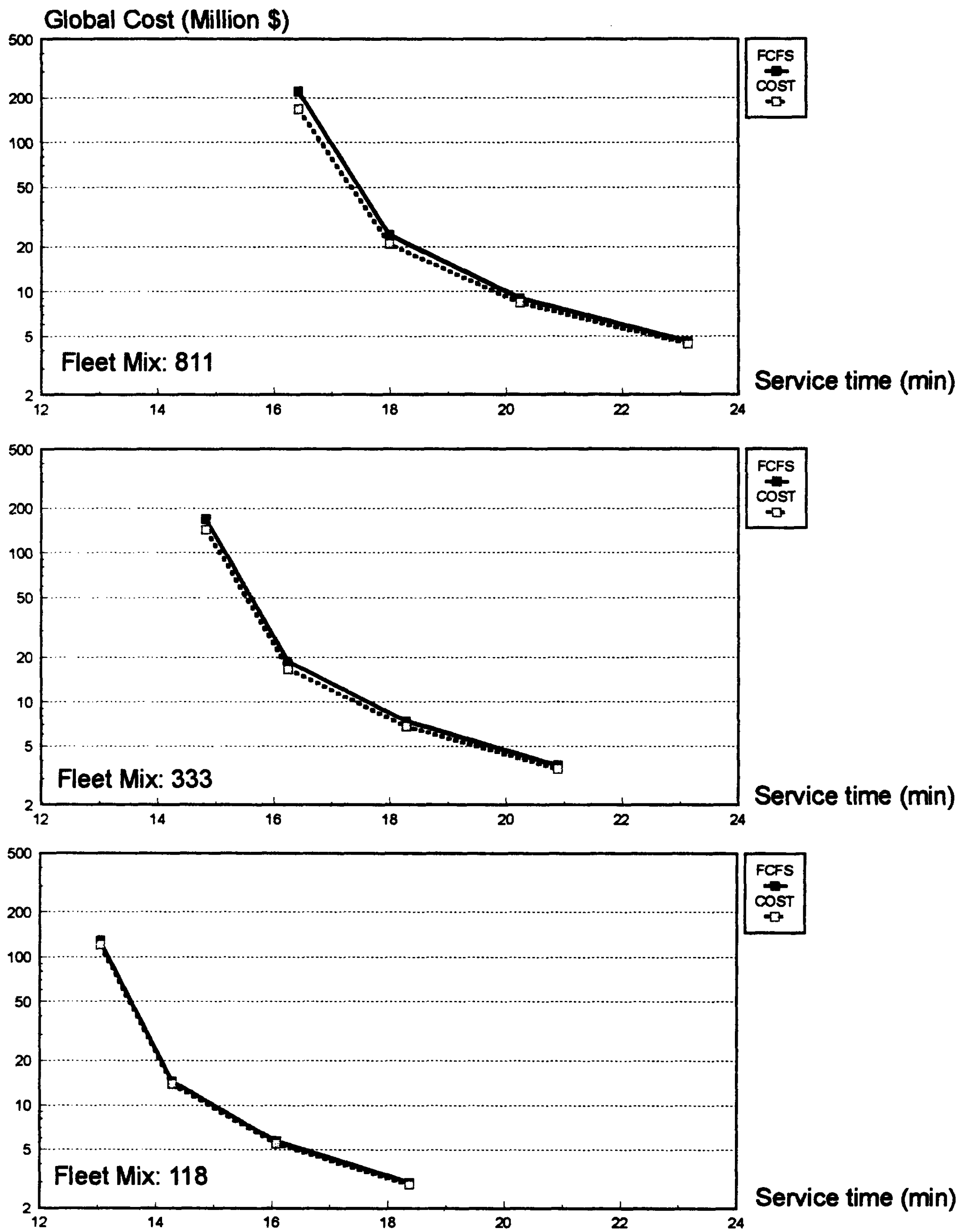
EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
MULTI SPEED - MAXIMUM DISTANCE
2 WAY, 10 NM



EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
MULTI SPEED - MINIMUM DISTANCE
2 WAY, 10 NM



EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
ONE SPEED - MAXIMUM DISTANCE
2 WAY, 10 NM



EFFECT OF QUEUEING ROUTINE IN GLOBAL COST
ONE SPEED - MINIMUM DISTANCE
2 WAY, 10 NM